
LOAD
REVISTA PARA USUARIOS DE
MSX
AÑO 1 Nº 8 ₳ 2,70 REP. ARGENTINA
SOFT NUEVO PARA TALENT, TOSHIBA Y SVI
EL CEREBRO DE LOS CIRCUITOS
PROGRAMAS INEDITOS
INTERFACE PARA CONECTAR ACCESORIOS
MANEJANDO LAS PANTALLAS
TOSHIBA HX-20 EN DETALLE
MSX
INS
DEL



A la Talent MSX nada le es imposible

El Club de Usuarios de MSX ya funciona en sus tres direcciones: CABILDO 2027 - 1º A CORDOBA 654 - P.B. (1054) y TUCUMAN 2044 - 1º - CAPITAL

Invitamos a los felices usuarios de la TALENT MSX al **curso gratuito** de introducción al fabuloso mundo de MSX.

Participe del Club de Usuarios de MSX y encuéntrese con sus amigos que también tienen la TALENT MSX, e intercambiará programas, datos y chimentos.

Podrá probar todos los accesorios de la línea MSX, ¡¡desde disketteras hasta robots!!

Podrá ver y leer todo lo que le interese sobre la norma MSX: catálogos, libros y revistas de todo el mundo. Todo con la seguridad, respaldo y seriedad que sólo TALENT puede brindarle.

¡Para inscribirse, no olvide traer su factura de compra!

ClubTalent MSX



AÑO 1 Nº8



Director General
Ernesto del Castillo

Director Editorial
Cristian Pusso

Director Periodístico
Fernando Flores

Director Financiero
Javier Campos Malbrán

Secretario de Redacción
Ariel Testori

Redacción
Eduardo Mombello
Andrea Sabin Paz

Arte y Diagramación
Fernando Amengual
Tamara Migelson

Departamento de Avisos
Oscar Devoto
Nelso Capello

Departamento de Publicidad
Guillermo González Aldalur

Servicios Fotográficos
Víctor Grubicy
Comesaña

Load Revista para usuarios de MSX es una publicación mensual editada por Editorial PROEDI S.A., Paraná 720, 5º Piso, (1017) Buenos Aires. Tel.: 46-2886 y 49-7130. Registro Nacional de la Propiedad Intelectual: E.T. M. Registrada. Queda hecho el depósito que indica la Ley 11.723 de Propiedad Intelectual.

Precio de este ejemplar: A 2,70
Impresión: Calcotam. Fotocromo tapa: Columbia. Fotocomposición: Interamericana Gráfica
Los ejemplares atrasados se venderán al precio del último número en circulación.

Distribuidor en Capital: Martino, Juan de Garay 358, P.B. Capital.
Distribuidor interior: DGP: Hipólito Yrigoyen 1450, Capital Federal. T.E. 38-9266/9800.

CORREO ARGENTINO CENTRAL | FRANQUEO PAGO CONCESION Nº 2538

Sumario

LA NORMA EDUCATIVA

Con la inauguración del Centro de Formación Docente en Informática la computación y la educación han dado un paso más para combinarse en bien de la sociedad.
El Centro organizado por la Fundación Nuestra Señora de la Merced y equipado por la empresa Telemática cuenta con diez equipos Talent MSX para la capacitación.
Que el Consejo Nacional de Educación Católica (Consudec) haya optado por computadores de la norma MSX es una muestra más de las bondades de estos equipos en el terreno del aprendizaje.
Y aún más, el Centro, seguro de las posibilidades futuras tanto del software como del hardware, se esforzará en el desarrollo de investigaciones sobre programas aplicables a la educación.
Cabe destacar, finalmente, el alto nivel de los profesionales que trabajan en este proyecto.

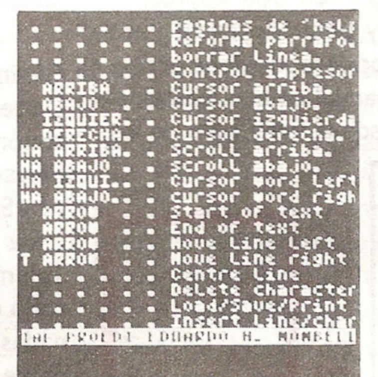
LOS EDITORES

UTIL PROGRAMA PARA ODONTOLOGOS



Los movimientos contables del consultorio, la distribución de citas a los pacientes o la actualización de una ficha siempre fueron tareas paralelas del dentista. Ahora, un sistema de gestión realiza estas "necesidades básicas" y alivia las tareas del profesional. (Pag.8)

MANEJANDO LAS PANTALLAS



¿Cómo y por qué podemos visualizar las letras y símbolos en la pantalla? Un tema muy interesante, pero que muy pocos conocemos. En esta nota hablaremos de la estructura de la parte de video de las computadoras, pero en especial de nuestras MSX. (Pag.14)

TOSHIBA HX-20, EN DATALLE

En estas páginas veremos los cuadros de especificaciones técnicas de esta nueva y poderosa máquina. (Pag.18)

EL CEREBRO DE LOS CIRCUITOS

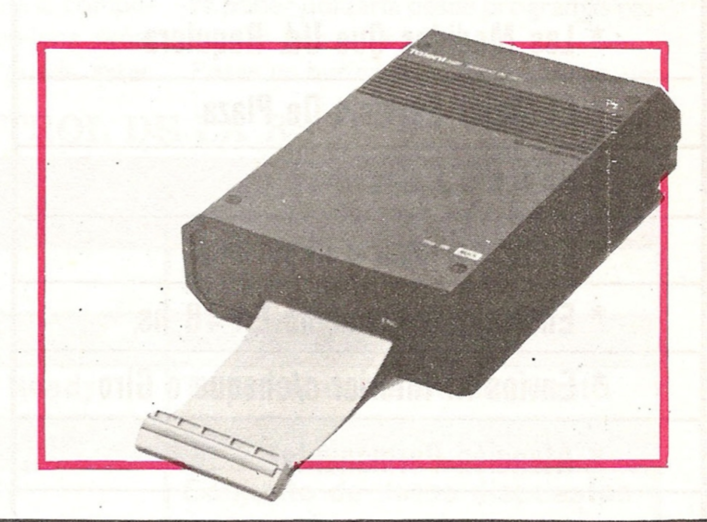
Es el encargado de llevar a cabo y organizar las funciones para las cuales fue preparada la computadora. Estos componentes no se encuentran sólo en la máquina, sino también en relojes, radios y otros artefactos electrónicos. (Pag.26)

LAS RIENDAS DEL ASSEMBLER (3ra. Parte)

Saber el significado de cada registro de este lenguaje es importante si no se quiere limitar la capacidad de trabajo. Por eso les explicamos uno por uno, y entre ellos el encargado de mostrarnos qué ocurre con nuestro conocido Z-80. (pág. 30)

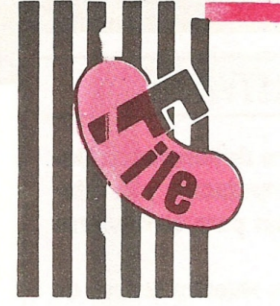
PROGRAMAS

SECCIONES FIJAS



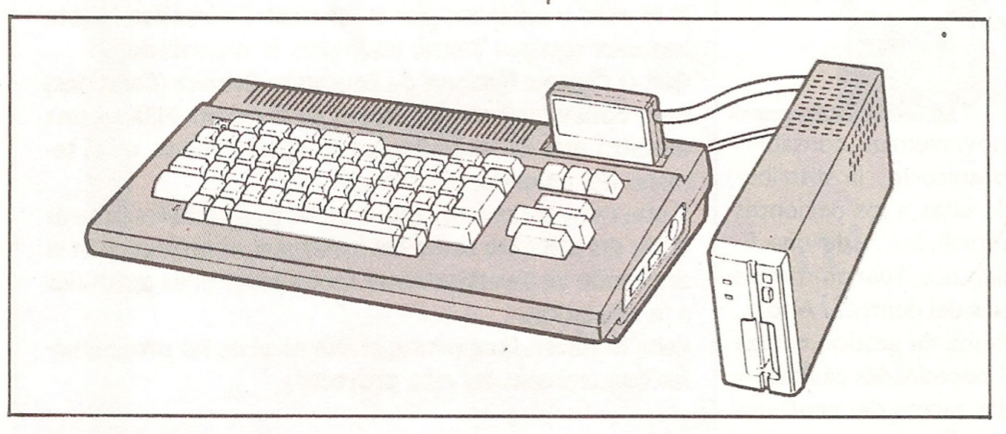
PARA CONECTAR ACCESORIOS

Les presentamos la interface que posibilita la conexión de la MSX con modems, impresoras u otra computadora. Se trata de un conector que tiene 25 agujas por medio de las que se establece la comunicación. (Pag. 5)



HX-F101

Este es el nombre del drive de discos que ya se encuentra en los comercios, y que podremos usar junto a la MSX de TOSHIBA.
Como ya hemos mencionado, posee la capacidad de formatear un disco de 3



pulgadas y media —que dicho sea de paso, sin ser los disquetes laser, son de la mejor tecnología en medios magnéticos que se han desarrollado para este tipo de computadoras— en 360 kbytes.
Este espléndido y pequeño drive y no por esto menos poderoso, viene acompañado por un manual en el que podremos aprender desde el significado de los archivos de programas, secuenciales, o aleatorios, hasta el manejo más sofisticado del DOS, pasando por el detalle de todas y cada una de las instrucciones que ofrece este magnífico sistema operativo para su manejo. Según personal de Toshiba todos los periféricos, por responder estrictametne a la norma MSX, son totalmente compatibles con cualquiera de la norma.

NUEVAS DIRECCIONES

El Club de Usuarios MSX y el Centro para el Desarrollo de la Inteligencia —CEDI— informan cambios en sus direcciones. Las mismas son: Cabildo 2027, Piso 1°; Córdoba 654 P.B., y Tucumán 2044, Piso 1°, de Capital Federal.

CONCURSO *Load MSX*

Dada la gran cantidad de programas que fueron enviados, y su calidad, el jurado ya está tomando una temperatura no precisamente comparable a la del Polo, sino todo lo contrario.
Ardua será la tarea de estos profesionales que harán posible que en nuestra próxima publicación podamos mostrar a los programadores favorecidos y sus obras.

TALENT Y LA EDUCACION

Fue inaugurado el Centro de Formación Docente en Informática con la organización y el equipamiento de la Fundación Nuestra Señora de la Merced y Telemática S.A. El Centro, ubicado en Esmeralda 761, está dotado de 10 equipos Talent MSX y capacitará a docentes de todo el país con dictados de cursos y jornadas a cargo de especialistas. Por otra parte, se desarrollan investigaciones sobre programas aplicables a la educación y se instrumentará la red nacional de establecimientos educativos del Consejo Nacional de Educación Católica (Consudec). Al acto de inaug ıración asistieron el secretario del Consudec, hermano Septimio Walsh, el director de Telemática S.A. Carlos Manzanedo, autoridades de la educación y docentes.



PARA CONECTAR ACCESORIOS

Les presentamos la interface que posibilita la conexión de la MSX con modems, impresoras u otra computadora. Se trata de un conector que tiene 25 agujas por medio de las que se establece la comunicación.

Todas las computadoras permiten que se les incorporen distintos periféricos como impresoras, disqueteras o al menos un grabador de cinta. Para comunicarse con el exterior, cuentan con entradas especiales para joystick tipo Atari, video, etcétera.
Las MSX en particular se pueden conectar con otros periféricos como robots y modems telefónicos.

A las MSX también se les puede agregar la interface RS-232. Se trata de un conector de una una de las normas estándar más comunes para las comunicaciones entre la computadora y periféricos, que tiene 25 agujas por medio de las cuales se establece la comunicación.

La transmisión de los datos puede ser unidireccional o bidireccional, dependiendo del artefacto con el cuál esté conectada la computadora.
Los datos en nuestras MSX se pueden transmitir de un punto a otro de dos formas: como un solo bloque de 8 bits en paralelo, enviados por 8 hilos, o bien en forma serial. Cuando se transmiten de esta última manera, los datos pasan por un solo hilo y cada bit del octeto sigue al otro.
Obviamente la transmisión en paralelo de los datos es más veloz que la serial. Pero sin embargo, en ocasiones es conveniente utilizar el pasaje de datos serialmente porque sólo se necesita de un hilo.
Por ejemplo, la mayoría de las impresoras utilizan el modo paralelo para enviar información y en cambio los modems suelen manejar los datos serialmente.
Si se emplea esta última forma de transmisión de datos, es necesario separar cada bit que pasa por el hilo. La forma más común, es enviar un código especial entre bit y bit, para que el equipo receptor pueda distinguir entre dato y dato.
Son pocas las MSX que traen incorporada esta potente interface, pero se les puede agregar fácilmente conectándola, según el modelo, por ejemplo en la ranura destinada a los cartuchos. Como aclaramos al comienzo, la RS-232 cuenta con 25 agujas o "caminos".

FUNCIÓN DE LOS CAMINOS DE LA RS-232

De las 25 agujas conectoras disponibles en esta interface, comentaremos la función de las nueve más importantes:
* RI: es la sigla de las palabras Ring Indicator. No es utilizada en la mayoría de las MSX, pero está incorporada en los modems telefónicos que realizan la contestación automática de llamadas.
* FG: proveniente de las palabras Frame Ground. No es otra cosa que la descarga tierra.
* DCD: Data Carrier Detect. Se emplea al establecerse una comunicación entre dos modems telefónicos. A través de este medio se envía una señal desde un modem, para avisarle a la computadora que está recibiendo una señal desde otro modem. Este puede estar ubicado a distancia y preparado para enviar datos.
* DSR: Data Set Ready. Por medio de este hilo se avisa a la computadora que el periférico conectado está listo para recibir los datos.
* DTR: Data Terminal Ready. Sirve para indicar que se encuentra dispuesto el conector RS-232.
* CTS: Clear To Send. La señal en este camino se puede emplear para evitar que la computadora trasmita los datos.
* RTS: Ready To Send. Con este hilo se puede impedir que el periférico devuelva datos al ordenador.
* TSD: (Trasmit Data) y RXC (Receive Data) son utilizados cuando se conecta la computadora con algún artefacto que le pueda devolver información al ordenador.
El flujo de datos se puede realizar en el "modo medio duplex", es decir la transmisión de datos en una sola dirección, y en "modo duplex pleno" que es cuando se reciben y trasmiten los datos simultáneamente en dos direcciones.
El resto de los caminos controlan la función entre los periféricos y el ordenador con señales ON/OFF.

CARACTERÍSTICAS DE LA RS-232 C.

La empresa Talent desarrollo la interface RS-232 C TRX-700. Esta interface incorpora el software necesario para poder utilizarla desde programas realizados por el usuario.
Posee un buffer de 128 caracteres, una

LINEAS DE CONTROL DE LA RS-232

CAMINOS	FUNCIÓN
RXD - Receive Data	Recibe datos
RTS - Ready to Send	Dispuesto para enviar
DTR - Data Terminal Ready	Terminal de datos dispuesto
DCR - Data Carrier Detect	Detector portador de datos
DSR - Data Set Ready	Conjunto de datos dispuestos
TXD - Trasmit Data	Trasmisor de datos
CTS - Clear to Send	Libre para el envío
FG - Frame Ground	Descarga tierra
RI - Ring Indicator	Indicador de llamada



opción de corte de transmisión de datos por exceso de tiempo (timeout) y permite además, controlar la velocidad de pasaje de los datos, que puede ser desde 50 hasta 19200 baudios.
Con esta interface se pueden establecer protocolos del tipo CTS o XON/XOFF.

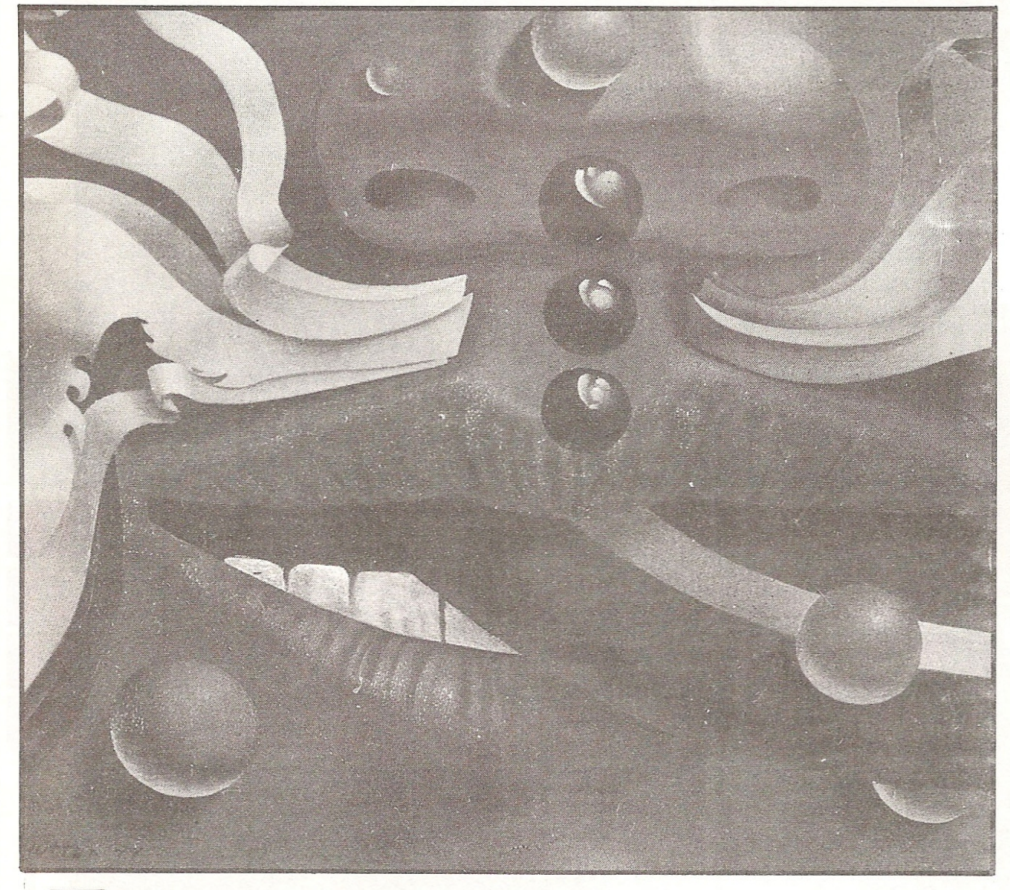
Otra de las sorprendentes ventajas de esta entrada/salida RS-232 C es la posibilidad de procesar las comunicaciones por medio de interrupciones por evento. En otras palabras, si el programa está ejecutando cierta tarea y al mismo tiempo recibe un carácter enviado desde otro modem, pasará a realizar otra acción para responder al mensaje recibido.

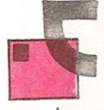
ESPECIFICACIONES TÉCNICAS DEL RS-232 C

Este cartucho se conecta directamente al bus de expansión. La velocidad de transmisión de datos puede ser: 50, 75, 110, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 7200, 9600 e inclusive 19200 baudios.
Otra ventaja es que permite el procesamiento de las comunicaciones desde Basic.
La capacidad del buffer es de 128 caracteres como mencionamos anteriormente.
Por último aclaramos que esta interface trae memoria opcional para incorporar una planilla electrónica: MSX-PLAN, un procesador de textos: MSX-WRITE y software de aplicación del usuario.
Como vimos, esta interface es un poderoso puente entre nuestra MSX y el exterior, permitiéndonos aprovechar hasta el último bit de la computadora.

UTIL PROGRAMA PARA ODONTOLOGOS

Los movimientos contables del consultorio, la distribución de citas a los pacientes o la actualización de una ficha siempre fueron tareas paralelas del dentista. Ahora un sistema de gestión realiza estas "necesidades básicas" y alivia las tareas del profesional.





Cada vez que vamos al dentista, sólo vemos una parte de la tarea que este profesional realiza dentro del consultorio.
Los odontólogos no sólo se dedican a curar dientes sino que también deben realizar tareas ajenas a su profesión para dirigir el consultorio.
No es sencillo distribuir los turnos a los pacientes, como tampoco llevar un control con las obras sociales y al mismo tiempo, una ficha médica por paciente. Se debe ser muy ordenado para que el consultorio funcione correctamente y no tener en algún momento cuatro pacientes a la vez, esperando ser atendidos.

ODONTOPACK es un sistema de gestión, que cumple estas necesidades básicas del odontólogo.
Principalmente, sus funciones son:
* controlar los movimientos contables del consultorio, como cobros, pagos, deudas, etcétera.
* llevar una ficha odontológica con registro gráfico de los tratamientos realizados.
* actualiza la agenda de citas del profesional, llevando estadísticas de las citas concertadas por semana y por día.

En síntesis, se abarcan los principales puntos que debe desarrollar un dentista para administrar su clínica dental.
Este sistema está excelentemente presentado en una carpeta. En su interior se encuentra un manual de instrucciones, junto con dos discos y un cartucho. Todos estos elementos forman ODONTOPACK.
La firma HIPOCAMPO INFORMATICA S.R.L. es la creadora de este útil sistema. Además, es el primer soft disponible para los odontólogos poseedores de una MSX y la firma Cerveux lo distribuye.
Para la creación de este conjunto de programas de gèstión, Hipocampo se basó en tres detalles fundamentales:
* que el odontólogo, en general, no posee conocimientos profundos de la técnica contable,
* que la generalidad de los dentistas no tiene experiencia alguna en computación, y por último
* que varias de las tareas previstas en el ODONTOPACK puedan ser llevadas a cabo por la asistente del profesional.

Respetando estos tres puntos, el sistema se maneja fácilmente, sin ser necesario, prácticamente, recurrir al manual de instrucciones.

CAPACIDAD

Por medio de este sistema, podremos llevar un registro de:
* 500 pacientes,
* 50 obras sociales,
* 30 cuentas contables,
* 450 eventos mensuales, y
* 200 órdenes de mecánico.

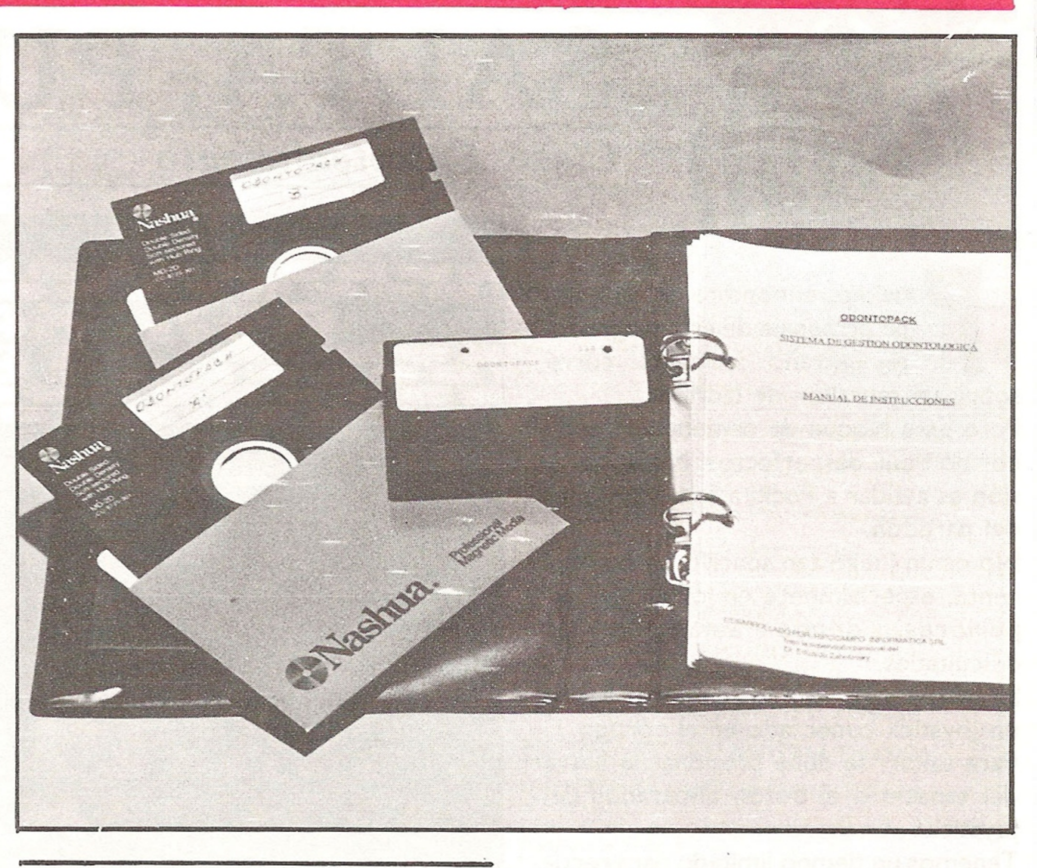
MANUAL DE INSTRUCCIONES:

Además de lo fácil que es trabajar con este sistema, —sencillo de aprender— el mismo viene acompañado por un claro manual.
En el mismo, se tratan los siguientes temas:
* una introducción al ODONTOPACK,
* un resumen del sistema donde se ven globalmente las posibilidades de este soft,
* se explica cómo comenzar a utilizarlo, dando un ejemplo. Se detallan paso a paso las operaciones a realizar para poner en marcha el sistema,
* la forma de sacar provecho a cada uno de los tres módulos que componene a ODONTOPACK: modulo I- contabi-

lidad del consultorio; módulo 2- ficha odontológica; módulo 3- agenda de citas.
Para lograr entender fácilmente el empleo de este utilitario, Hipocampo agregó un anexo (A) con datos para tomar como ejemplo en las explicaciones.
Las operaciones que permiten manejar el sistema son rápidamente asimiladas, sin necesidad de tener experiencia en el manejo de otro sistema similar. Esto principalmente se debe a que la explicación de cada ítem es clara y concisa. Inclusive les enseña a los que jamás han tenido previamente una MSX, cómo se enciende y se conecta.

UTILIZACION DEL ODONTOPACK

El sistema se caracteriza por contar con abundante cantidad de pantallas y por la rapidez con que maneja los archivos.
Al comenzar a trabajar, nos encontramos frente a un menú principal con las siguientes opciones:
1 - Datos del profesional
2 - Anotación/consulta de ficheros
3 - Balance de cuentas
4 - Ingreso de eventos del mes
5 - Consulta de eventos
6 - Consulta de ficha odontológica
7 - Agenda de citas
8 - Cierre del mes
9 - Fin de trabajo
Con esta pantalla sencilla de interpretar y completa, podemos acceder a todas las opciones que nos brinda el utilitario.
Como se puede advertir en este menú, se puede llevar el registro con la información referida al profesional, a los pacientes, a cada obra social, a cuentas contables y a órdenes de trabajos encargadas a mecánicos.
Las ventajas de este sistema son:
* permitir actualizar los datos de cada una de las fichas mencionadas anteriormente,
* incorporar eventos acaecidos en el consultorio en relación a esas fichas,
* consultar por pantalla datos de las fichas y de los eventos del mes en relación a cada una de ellas, y también
* efectuar las mismas consultas impresas en papel.
Cada vez que se nos requiera la entrada de un dato, aparecerán en pantalla los mensajes aclaratorios, que nos guiarán en la forma en que se debe entrar la información.
Reiteramos que se trata de un sistema construido para ser utilizado sin la necesidad de contar con conocimientos de computación.



FICHA ODONTOLOGICA

Se trata de una representación gráfica en pantalla de las piezas dentales de cada paciente.
Sobre cada una de ellas se puede representar cada tratamiento realizado, como la eventual ausencia de la pieza.
Manejar esta parte es sumamente sencillo. Sólo se deben seguir las instrucciones que la misma computadora nos va formulando. Inclusive nos aclara cuáles son las posibles respuestas a ingresar, para evitar confusiones.
Primeramente, el ordenador preguntará por el código del paciente, (los pacientes están representados por códigos para agilizar el funcionamiento del sistema).
Se puede incluir diversos tratamientos como ortodoncia, endodoncia, tratamientos de conducto, obturación con amalgama, etcétera.
Una vez detallado el tratamiento, podremos ver la ficha médica de ese paciente, actualizada.
Sólo debemos seguir las instrucciones que nos va dando la computadora.

AGENDA DE CITAS

No es sencillo llevar una agenda para un consultorio si pretendemos distribuir correctamente los horarios de atención.
Para esta agenda, el año está dividido en 53 semanas de cinco días hábiles cada una.
Por cada día, se pueden registrar hasta 16 citas.

Veamos un ejemplo de este interesante módulo.
Una vez que hayamos conseguido ingresar en este módulo (respondiendo con "7" al menú principal) y cambiar el disco como lo pide la computadora, se debe entrar el mes a consultar.
Entonces aparecerá el calendario correspondiente al mes ingresado.
Se verán, además, los números de semanas y la cantidad de pacientes citados en cada una de ellas.
Inmediatamente, se preguntará al usuario por el número de la semana que desea ver. Y aparecerá cada uno de los cinco días hábiles de la semana ingresada, y los pacientes citados por cada día.
Ahora, el ordenador preguntará qué día se quiere consultar.
Si nos equivocamos al ingresar algún dato, o bien consideramos que mejor hubiera sido entrar otra información, se arregla simplemente presionando RETURN. De esta manera pasaremos a la pantalla anterior a la que nos encontramos, volviéndonos a preguntar la computadora como cuando nos equivocamos, pero esta vez para corregir.
Al ingresar el número del día, y siguiendo los pasos que nos va marcando el sistema, aparecerán las citas realizadas para ese día.

La información en este módulo es: "hora de la cita", "nombre del paciente".
A esta agenda se le pueden agregar citas, consultarlas simplemente, o cancelarlas, actualizando automáticamente todas las variables relacionadas con estos datos.



SALTA, PUCKI

Clase: Entretenimiento

Pucki es el nombre del simpático personaje de este juego. Este extraño animalito corre sobre un murallón de ladrillos.

Pero este bloque de cemento y barro cocido tiene desperfectos. Nuestra misión es ayudar a Pucki a saltar las fallas del paredón.

No es un juego tan sencillo como aparenta, especialmente en los niveles superiores, donde aumentan las dificultades.

Se puede jugar con los cursores o con un joystick conectado en el pórtico 1. Para saltar, se debe presionar la barra del espacio o el botón disparador del joystick.

Tenemos un tiempo limitado para recorrer el paredón sin caernos. Sólo cuando consigamos recorrerlo exitosamente, pasaremos de nivel.

En el ángulo superior izquierdo aparecerá el puntaje de nuestro juego, y en el derecho, el mayor puntaje alcanzado desde que comenzó el juego.

Se trata de sumar la mayor cantidad de puntos con sólo tres vidas.

Para recorrer el paredón, debemos movernos hacia el costado con los cursores o con el joystick.

VARIABLES IMPORTANTES:

SC: puntaje
HS: record alcanzado
RO: nivel
RE: vidas
Y: coordenadas de la fila del sprite
T: tiempo

ESTRUCTURA DEL PROGRAMA

10-50: inicialización de variables
60-90: sonido del comienzo
100-310: parte principal del programa
320-350: aumenta el puntaje si salto el pozo
360-370: dibuja pared
380-390: hace saltar al sprite
400-410: imprime tiempo
420-430: hace desaparecer al sprite
440-460: maneja la pausa
470-490: busca record
500-510: imprime cantidad de vidas
520-530: imprime nivel
540-560: dibuja la pantalla del juego
570-630: rutina de perder
640-700: reinicializa variables
710-750: fin del juego
760-810: fin del nivel y recompensa
820-970: pantalla inicial
980-1110: define sprite, juego de caracteres y rutina Assembler
1120: datos de los colores
1130-1150: dato para los caracteres
1160-1180: datos para definir sprites
1190-1200: datos para la rutina Assembler

```
10 SCREEN 1,0,0:COLOR 15,1,1:WID
TH 32:KEY OFF:A=RND(-TIME)
20 CLEAR 400,&HDFFF:DEFINT A-Z:D
IM C(7):GOSUB 980
30 HS=0:P$="s10m10000t120132o6"
40 GOSUB 820
50 RE=2:RO=1:SC=0:TT=500:R=10
60 GOSUB 640
70 PLAY P$+"DEDEDEFGAB":GOSUB 52
0
80 FORI=1 TO1500:NEXT
90 IF (STICK(0)AND STRIG (0)) OR
 (STICK(1) AND STRIG(1)) THEN M=1
 ELSE M=0
100 '
110 X=0:Y=17:XX=0:JP=0:JH=0:T=TT
:POKE &HE000,X:POKE &HE001,0
120 GOSUB 540:GOSUB 360:GOSUB 40
0
130 S=STICK(0) OR STICK(1):ST=ST
RIG(0) OR STRIG(1)
140 IF JP GOTO 170
150 IF ST THEN JP=1:JH=2:PLAY P$
+"EG":GOTO 170
160 IF S=1 OR S=2 OR S=8 THEN JP
=2:JH=3:PLAY P$+"DF"
170 IF JP=1 THEN IF ST THEN Y=Y-
1 ELSE JP=-1
180 IF JP=2 THEN IF S=1 OR S=2 O
R S=8 THEN Y=Y-1 ELSE JP=-2
190 IF 17-Y>JH THEN JP=JP*-1
200 IF JP<0 THEN Y=Y+1:IF Y>16 T
HEN JP=0:JH=0
210 IF S>1 AND S<5 THEN XX=XX-(X
X<3):GOTO 240
220 IF S>5  THEN XX=XX+(XX>-3):G
OTO 240
230 XX=XX-(SGN(XX))
240 IF ABS(XX)>1 THEN X=X+SGN(XX
):X=X-(X<0)
250 GOSUB 380
260 POKE &HE000,X:GOSUB 360
270 GOSUB 320
280 IF X>254 GOTO 760
290 T=T-1:GOSUB 400:IF T<1 GOTO
580
300 I$=INKEY$:IF I$=CHR$(13) THE
N GOSUB 440
```

```
310 GOTO 130
320 V=VPEEK(6735):IF M GOTO 350
330 IF JP=0 AND V=104 GOTO 570
340 IF V=104 AND ABS(JP)<2 THEN
SC=SC+1:GOSUB 470
350 RETURN
360 DEF USR=&HE100:A=USR(0)
370 RETURN
380 PUT SPRITE 1,(120,Y*8-1),7,1
390 RETURN
400 LOCATE 13,3:PRINT USING" TIE
MPOa###";T;
410 RETURN
420 PUT SPRITE 1,(0,209)
430 RETURN
440 PLAY P$+"L64FAFA":LOCATE 13,
21:PRINT"PAUSA":
450 IF INKEY$=CHR$(13) THEN LOCA
TE 13,21:PRINT"aaaaa";:RETURN
460 GOTO 450
470 IF SC>HS THEN HS=SC
480 LOCATE 0,2:PRINT USING"PUNAT
JEa####0a MAYOR PUN.a####0";SC;HS
;
490 RETURN
500 LOCATE 30-LEN(STRING$(RE,"p"
)),3:PRINT STRING$(RE,"p");
510 RETURN
520 CLS:LOCATE 12,9:PRINT USING"
NIVEL ##";RO;
530 RETURN
540 CLS:LOCATE 0,1:PRINT STRING$
(128,"a");
550 GOSUB 470:GOSUB 500:LOCATE 0
,18:PRINT STRING$(160,"a");
560 RETURN
570 PUT SPRITE 1,(120,Y*8-1),7,2
:GOTO 590
580 PUT SPRITE 1,(120,Y*8-1),7,3
590 LOCATE 13,10:PRINT"PERDISTE!
!";
600 PLAY P$+"BGEC":FOR I=1 TO 10
00:NEXT
610 GOSUB 420:FOR I=1 TO 3000:NE
-XT
620 RE=RE-1: IF RE<0 GOTO 710
630 GOTO 70
640 E=&HE200:FOR I=1 TO 32:POKE
·E,97:E=E+1:NEXT
650 FOR I=1 TO 223
660 IF I MOD 6=0 THEN A=97:GOTO
680
670 IF INT(RND(1)*R) THEN A=97 E
LSE A=104
680 POKE E,A:E=E+1:NEXT
690 FOR I=1 TO 32:POKE E,97:E=E+
1:NEXT
700 RETURN
710 LOCATE 13,10:PRINT SPC(10);:
LOCATE 11,11:PRINT"FINALIZO EL JU
EGO";
720 PLAY P$+"T140L16FF+GG+8AA+8B
B8G07C8.06G807CC4"
730 FOR I=1 TO 1500:IF STRIG(0)
OR STRIG(1) THEN I=1500
740 NEXT:GOSUB 420
750 GOTO 40
760 PLAY"S3M2000T150L1606EFFF+G8
EF8DE8CD805B06C8.07C806C4"
770 LOCATE 8,9:PRINT"VAMOS TODAV
IA !";
780 IF T<1 GOTO 800
790 FOR I=1 TO T:T=T-1:GOSUB 400
:SC=SC+1:GOSUB 470:NEXT
800 GOSUB 420:RO=RO+1:TT=TT+(TT>
300)*50:R=R+(R>5)
810 GOTO 60
820 CLS
830 PRINT"        aaa  a   a    aaa
 a"
840 PRINT"       a    a a a      a
a a"
850 PRINT"        a   aaa a      a
aaa"
860 PRINT"         a a a a       a
a a"
870 PRINT"       aaa a a aaa    a
a a"
880 PRINT:PRINT
890 PRINT"        aa   a a aaa a a
 a"
900 PRINT"        a a a a a    a a
 a"
910 PRINT"        aa   a a a    aa
 a"
920 PRINT"        a    a a a    a a
 a"
930 PRINT"        a    aaa aaa a a
 a"
940 LOCATE 2,17:PRINT"PREIONA LA
 BARRA DE ESPACIO";
950 IF (STRIG(0) OR STRIG(1))=0
GOTO 950
960 PLAY"S3M2000T150L1606CEGDFEO
7C1"
970 RETURN
980 FOR I=0 TO 7:READ Q$:C(I)=VA
L("&H"+Q$):NEXT
990 FOR I=1 TO 3:READ A,Q$:FOR J
=0 TO 2:FOR K=0 TO 7
1000 VPOKE J*&H800+A*8+K,VAL("&H
"+MID$(Q$,K*4+1,2))
1010 VPOKE &H2000+J*&H800+A*8+K,
VAL("&H"+MID$(Q$,K*4+3,2))
1020 NEXT K,J,I
1030 FOR I=0 TO 2:FOR J=&H100 TO
 &H2FF
1040 V=PEEK(7103+J):VPOKE I*&H80
0+J,V OR V/2
1050 IF J<&H180 OR J>&H1CF THEN
A=C(J MOD 8) ELSE A=&HF1
1060 VPOKE &H2000+I*&H800+J,A:NE
XT J,I
1070 FOR I=1 TO 3:S$="":FOR J=1
TO 8:READ Q$
1080 S$=S$+CHR$(VAL("&H"+Q$)):NE
XT:SPRITE$(I)=S$:NEXT
1090 FOR I=&HE100 TO &HE111:READ
Q$:POKE I,VAL("&H"+Q$):NEXT
1100 DEF USR=&H7E:A=USR(0)
1110 RETURN
1120 DATA 41,51,71,F1,F1,71,51,4
1
1130 DATA 97,7E917E817E610061E79
1E781E7610061
1140 DATA 112,0071667124717E71EB
71FF716671C371
1150 DATA 104,00F700F720F750F704
F70AF700F700F7
1160 DATA 00,66,24,7E,EB,FF,66,C
3
1170 DATA 00,00,00,42,24,81,42,0
0
1180 DATA 42,24,00,C3,00,24,42,0
0
1190 DATA 21,00,E2,ED,4B,00,E0,0
9,11
1200 DATA 40,1A,01,20,00,CD,5C,0
0,C9
```



DESPERTANDONOS CON MUSICA

CLASE: UTILITARIO

Ahora, nos podremos despertar por las mañanas escuchando una agradable sinfonía. Se trata de un reloj que, además, a la hora que querramos nos despertará.

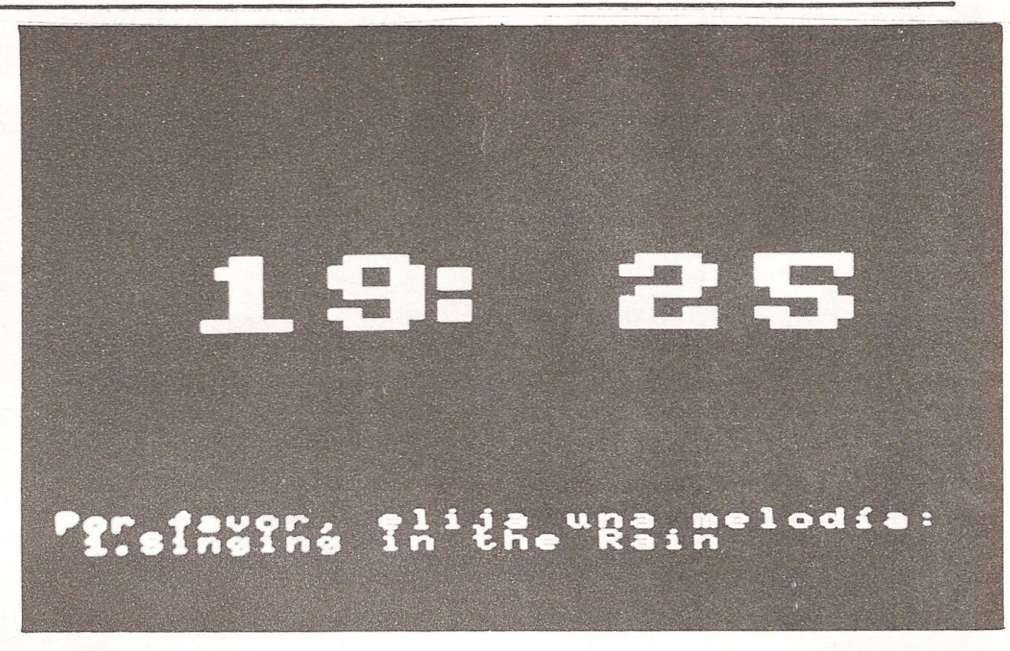
En este programa están almacenadas cinco melodías diferentes. Y es sencillo agregarle más.

Una vez copiado el programa y verificado, al ejecutarlo nos preguntará la hora en que nos queremos despertar. Luego nos pedirá que sincronicemos nuestro reloj pulsera con el de la computadora. No es complicado manejarlo; sólo es necesario seguir las instrucciones que el programa nos va dando.

Las teclas de función están definidas para ayudarnos a dirigir el programa:

F1 - Cambia la hora de despertarnos
F2 - Cambia. Sincroniza los relojes
F3 - Elige melodía (con la tecla de cursor izquierda y derecha se escoge una)
F4 - Muestra/borra la hora en panatalla
F5 - Vuelve a correr el programa

Con la barra de espacio, se detiene la ejecución de la melodía.



VARIABLES IMPORTANTES:

MEL: melodía escogida (incrementada en 1)
SUENA: si su valor es 1, sonará la música. Si es 0, no.
NOM$: nombre de las melodías
D1, D2, E1, E2: forman la hora para despertarse
M1, M2, H1, H2: forman la hora real

ESTRUCTURA DEL PROGRAMA:

10-190: define sprites e inicializa variables
200-250: inicializa los relojes
260-270: comienza a funcionar el reloj
280-300: chequea la hora real con la del despertador (porque si son iguales hace sonar la música)
310-330: actualiza la hora y los segundos
340-360: rutina para aceptar los datos
370-450: inicializa las variables del despertador
460-600: inicializa las variables del reloj
610-670: rutina para elegir melodía
680: desactiva la música
690-700: ejecuta la música
710-870: escribe o no los números de la hora en pantalla
880-920: elige desde que DATA se debe leer la música, según nuestra elección

930-1050: primera melodía
1060-1140: segunda melodía
1150-1230: tercera melodía
1240-1480: cuarta melodía
1490-1570: última melodía

```
10 KEY OFF:DEFUSR=&H90:DEFUSR1=&
H156
20 SCREEN 1,3:RESTORE
30 LOCATE 1,12:PRINT"ESPERA UN M
OMENTO, POR FAVOR"
40 FOR SP%=0 TO 10:FOR LAZO=1 TO
 32
50 READ D$:D%=VAL("&H"+D$)
60 SP$=SP$+CHR$(D%):NEXT LAZO
70 SPRITE$(SP%)=SP$:SP$="":NEXT
SP%
80 DATA 3F,3F,F0,F0,F0,F0,F3,F3,
FC,FC,F0,F0,3F,3F,00,00,F0,F0,3C,
3C,FC,FC,3C,3C,3C,3C,3C,3C,F0,F0,
00,00
90 DATA 03,03,0F,0F,03,03,03,03,
03,03,03,03,3F,3F,00,00,C0,C0,C0,
C0,C0,C0,C0,C0,C0,C0,C0,C0,FC,FC,
00,00
100 DATA 0F,0F,3C,3C,00,00,0F,0F
,3C,3C,3C,3C,3F,3F,00,00,F0,F0,3C
,3C,3C,3C,F0,F0,00,00,3C,3C,FC,FC
,00,00
110 DATA 0F,0F,30,30,00,00,03,03
,00,00,3C,3C,0F,0F,00,00,F0,F0,3C
,3C,3C,3C,F0,F0,3C,3C,3C,3C,F0,F0
,00,00
120 DATA 03,03,0F,0F,33,33,C3,C3
,FF,FF,03,03,0F,0F,00,00,C0,C0,C0
,C0,C0,C0,C0,C0,FC,FC,C0,C0,F0,F0
,00,00
130 DATA 3F,3F,3C,3C,3C,3C,0F,0F
,00,00,3C,3C,0F,0F,00,00,FC,FC,0C
,0C,00,00,F0,F0,3C,3C,3C,3C,F0,F0
,00,00
140 DATA 0F,0F,3C,3C,3C,3C,3F,3F
,3C,3C,3C,3C,0F,0F,00,00,F0,F0,3C
,3C,00,00,F0,F0,3C,3C,3C,3C,F0,F0
,00,00
150 DATA 3F,3F,30,30,00,00,00,00
,03,03,03,03,03,03,00,00,FC,FC,3C
,3C,3C,3C,F0,F0,C0,C0,C0,C0,C0,C0
,00,00
160 DATA 0F,0F,3C,3C,3C,3C,0F,0F
,3C,3C,3C,3C,0F,0F,00,00,F0,F0,3C
,3C,3C,3C,F0,F0,3C,3C,3C,3C,F0,F0
,00,00
170 DATA 0F,0F,3C,3C,3C,3C,0F,0F
,00,00,3C,3C,0F,0F,00,00,F0,F0,3C
,3C,3C,3C,FC,FC,3C,3C,3C,3C,F0,F0
,00,00
180 DATA 00,00,00,00,03,03,03,03
,00,00,03,03,03,03,00,00,00,00,00
,00,C0,C0,C0,C0,00,00,C0,C0,C0,C0
,00,00
190 FOR N=0 TO 4:READ NOM$(N):NE
XT
200 DATA Singing in the Rain,Min
ue,Nana,Noche de paz,Chotis
210 CLS:GOSUB 370
220 GOSUB 460
230 GOSUB 580
240 ON MEL+1 GOSUB 2000,2010,202
0,2030,2040
250 LOCATE 0,21:PRINT"Pulse [RET
URN] para empezar"
260 K$=INKEY$:IF K$<>CHR$(13) TH
EN 260 ELSE ON INTERVAL=3000 GOSU
B 310:INTERVAL ON:LOCATE 0,21:PRI
NT SPACE$(28);
270 ON KEY GOSUB 370,460,580,110
0,10:FOR LLAVE=1 TO 5:KEY(LLAVE)
ON:NEXT LLAVE:ON STRIG GOSUB 900:
STRIG(0)ON
280 IF SUENA=1 THEN GOSUB 1000
290 IF M2=D2 AND M1=D1 AND H2=E2
 AND H1=E1 AND DES=1 AND SUENA=0
THEN ON MEL+1 GOSUB 2000,2010,202
0,2030,2040:SUENA=1
300 GOTO 280
```

```
310 M2=(M2+1) MOD 10:PUT SPRITE
0,(180,70),15,M2:IF M2<>0 THEN 33
0 ELSE M1=(M1+1) MOD 6:PUT SPRITE
 1,(146,70),15,M1:IF M1<>0 THEN 3
30
320 H2=-((H1<2)*(H2+1)MOD10)-((H
1=2)*(H2+1)MOD 4):PUT SPRITE 2,(7
2,70),15,H2:IF H2<>0 THEN 330 ELS
E H1=(H1+1)MOD 3:PUT SPRITE 3,(38
,70),15,H1
330 RETURN
340 A=USR1(0):LOCATE 0,22:X=1:D$
=SPACE$(L)
350 K$=INKEY$:IF K$="" THEN 350
ELSE IF K$=CHR$(8) AND X>1 THEN P
RINTCHR$(127);:X=X-1:MID$(D$,X,1)
=" ":GOTO 350 ELSE IF K$=CHR$(13)
 THEN LOCATE 0,22:PRINTSPACE$(L):
RETURN ELSE IF K$=CHR$(8) THEN 35
0 ELSE MID$(D$,X,1)=K$:PRINTK$;
360 IF X=L THEN LOCATE 0,22:PRIN
T SPACE$(L):RETURN ELSE X=X+1:GOT
O 350
370 LOCATE 0,21:PRINT"Hora de de
spertarse (HH/MM)?"
380 L=5:GOSUB 340:DH$=D$
390 LOCATE 0,21:PRINT SPACE$(50)
400 IF INSTR(DH$," ") THEN DES=0
:DH$="00/00" ELSE DES=1
410 D2=VAL(MID$(DH$,5,1))
420 D1=VAL(MID$(DH$,4,1))
430 E2=VAL(MID$(DH$,2,1))
440 E1=VAL(MID$(DH$,1,1))
450 RETURN
460 LOCATE 0,21:PRINT"Hora actua
l (HH/MM)?"
470 L=5:GOSUB 340:AH$=D$
480 IF INSTR(AH$," ") THEN A=USR
1(0):LOCATE 0,22:PRINT SAPCE$(L):
GOTO 470 ELSE LOCATE 0,21:PRINT S
PACE$(50)
490 M2=VAL(MID$(AH$,5,1))
500 M1=VAL(MID$(AH$,4,1))
510 H2=VAL(MID$(AH$,2,1))
520 H1=VAL(MID$(AH$,1,1))
530 PUT SPRITE 0,(180,70),15,M2
540 PUT SPRITE 1,(146,70),15,M1
550 PUT SPRITE 2,(72,70),15,H2
560 PUT SPRITE 3,(38,70),15,H1
561 LOCATE 13,9:PRINTCHR$(220)
562 LOCATE 13,10:PRINTCHR$(223)
563 LOCATE 13,11:PRINTCHR$(219)
570 RETURN
580 LOCATE 0,21:PRINT"Por favor,
 elija una melodía:":A=MEL
590 LOCATE 0,22:PRINT USING"##&&
";A+1;".";NOM$(A)
600 K$=INKEY$:IF K$="" THEN 600
610 IF K$=CHR$(28) THEN A=(A+1)
MOD 5
620 IF K$=CHR$(29) THEN IF A>0 T
HEN A=A-1:GOTO 640 ELSE A=4:GOTO
640
630 IF K$=CHR$(13) THEN MEL=A:LO
CATE 0,21:PRINT SPACE$(58);:RETUR
N
640 LOCATE 0,22:PRINT SPACE$(28)
;:GOTO 590
900 A=USR(0):SUENA=0:DES=0:RETUR
N
1000 READ A$: IF VAL(A$)<0 THEN
ON MEL+1 GOSUB 2000,2010,2020,203
0,2040:GOTO 1000
1010 READ B$,C$:PLAY A$,B$,C$:RE
TURN
1100 P=NOT P::IF P THEN 1200
1110 PUT SPRITE 0,(180,70),15,M2
1120 PUT SPRITE 1,(146,70),15,M1
1130 PUT SPRITE 2,(72,70),15,H2
1140 PUT SPRITE 3,(38,70),15,H1
1150 LOCATE 13,9:PRINTCHR$(220)
1160 LOCATE 13,10:PRINTCHR$(223)
1170 LOCATE 13,11:PRINTCHR$(219)
1180 RETURN
1200 PUT SPRITE 0,(180,70),4,M2
1210 PUT SPRITE 1,(146,70),4,M1
1220 PUT SPRITE 2,(72,70),4,H2
1230 PUT SPRITE 3,(38,70),4,H1
1240 LOCATE 13,9:PRINTCHR$(32)
1250 LOCATE 13,10:PRINTCHR$(32)
1260 LOCATE 13,11:PRINTCHR$(32)
1270 RETURN
2000 RESTORE 3000:RETURN
2010 RESTORE 4000:RETURN
2020 RESTORE 5000:RETURN
2030 RESTORE 6000:RETURN
2040 RESTORE 7000:RETURN
3000 DATA v7o4t200s13m10g4.,v7o3
t200s13m10r4.,v7o6t200s13m10r4.
3010 DATA o5g1e8d4c8,c2r8c8o2g2.
,r1.
3020 O4A1R8G4.,O3C2R8C8O2G2.,R4A
8R4A8G+4A8R4.
3030 DATA o5c1c8d4e8,o3c2r8c8o2g
2.,r1.
3040 DATA g1r8o4g4.,o3c2r8c8o2g2
.,r4d8e8g8a8o7c8r4r4.
3050 DATA o5c1d8e4g2.e2g4.,o3c2r
8c8o2g2.o3c2r8c8o2g4.o3c+4.,r4.c4
c8r2.r4.o6g4g8r2.
3060 DATA g2.e4.d4o4a1.r8,d2r8d8
e4.f4d2.d8o2g2.,o6r1.r4a8g+4g8r2.
3070 DATA o5r4g2e8r4d4o4a1.r8,o3
d2r8d8o2g2.o3d2r8d8o2g2.,r1.r4d8c
+4d8r2.
3080 DATA r4o5g2g4e8d4o4a1r4g4.,
o3d2r8d8o2g2.o3d2r8d8o2g2.,r1.r8a
16a+16b2a+4.a4g8
3090 DATA o5g4g8r2.o4g4.,o3d2r8d
8o2g2.,r2.f+4g8r4.
3100 DATA o5d4d8r4.e4.o4g4o5c1.c
1c4r4.,o3d2r8o2g8a4.b4o3c2.c8o2g2
.,r2.g4.f4e4.c8d4o5a8o6c4.o5a4o6c
8e4c8d4o5a8o6c4.r4.
3110 DATA e4c8d4o4a8o5c4.o4a4o5c
8e4c8d4o4a8o5c4r8,o3c2r8c8o2g2.o3
c2r8c8o2g4r8,e4c8d4o5a8o6c4.o5a4o
6c8e4c8d4o5a8o6c4r8
3120 DATA -1
4000 DATA t120s9m4000o7c16o6b16o
7c16d16,t120s9m4000r4,t120s9m4000
r4
4010 DATA t120s9m16000c8o6c4e4g8
,t120s9m16000o4r8e4g4o5c8,t120s9m
16000o4r8c4e4g8
4020 DATA g8f8f4s9m4000f16e16f16
g16,o3b8d8o4g2,o3g8b8o4d2
4030 DATA s9m16000f8o5g4o6d4f8,r
8o3b4d4o4g8,r8o3g4b4o4d8
4040 DATA f8e8e4o7c8.o6a16,e8g8o
5c4r4,c8e8g4r4
4050 DATA s9m8000g8f+8f+8f+8o7c8
.o6a16,s9m4000r8f+8f+8f+8r4,s9m80
00r8d8a8d8r4
4060 DATA s9m8000g8f+8f+8f+8o7c8
.o6a16,s9m4000r8f+8f+8f+8r4,s9m80
00r8d8a8d8r4
4070 DATA b8g8e8o7c8s9m32000o6a8
..s9m4000g64f+64s9m32000g4r4,d4e4
f+4s9m32000g4r4,g4c4d4s9m32000g4r
4
4080 DATA -1
5000 DATA v12t100s9m8000o6e8f8,t
100r4,t100r4
5010 DATA g4r4e8f8,s9m8000o5g4e4
c4,s9m8000v12o4c4r2
5020 DATA g4r4e8f8,g4e4c4,c4r2
5030 DATA o7c4o6b4a4,g4e4c4,c2r4
5040 DATA a4g4d8e8,d4o4b4g4,v14o
3g4r2
5050 DATA f4r4d8e8,o5d4o4b4g4,g4
r2
5060 DATA f4r4d8f8,o5d4o4b4g4,g4
r2
5070 DATA b8a8g4b4o7c4r4,o5f4e4d
4c4r4,g4a4b4v12c4r4
5080 DATA -1
6000 DATA o5t120s9m8000g4.a8g4,o
4t120s9m8000g4e4c4,o3t120s9m16000
c2.
6010 DATA e4r2,g4e4c4,c2.
6020 DATA g4.a8g4,g4e4c4,c2.
6030 DATA e4r2,g4e4c4,c2.
6040 DATA o6d2d4,d4o3b4g4,o2g2.
6050 DATA o5b4r2,o4d4o3b4g4,g2.
6060 DATA o6c2c4,o4g4e4c4,o3c2.
6070 DATA o5g4r2,g4e4c4,c2.
6080 DATA a2a4,o5c4o4a4f4,f2.
6090 DATA o6c4.o5b8a4,o5c4o4a4f4
,f2.
6100 DATA g4.a8g4,g4e4c4,c2.
6110 DATA e4r2,g4e4c4,c2.
6120 DATA a2a4,o5c4o4a4f4,f2.
6130 DATA o6c4.o5b8a4,o5c4o4a4f4
,f2.
6140 DATA g4.a8g4,g4e4c4,c2.
6150 DATA e4r2,g4e4c4,c2.
6160 DATA o6d2d8,d4o3b4g4,o2g2.
6170 DATA f4.d8o5b4,o4d4o3b4g4,g
2.
6180 DATA o6c4r2,o4g4e4c4,o3c2.
6190 DATA e4r4,g4e4c4,c2.
6200 DATA o6c4.o5g8e4,g4e4c4,c2.
6210 DATA g4.f8d4,d4o3b4g4,o2g2.
6220 DATA c4r2,o4c4e4g4,o3c2.
6230 DATA o6c4r2,o5r2,c2.
6240 DATA -1
7000 DATA o6t80s9m4000g8e8,t80s9
m4000r4,t80s9m4000r4
7010 DATA g4g8e8g4g8e8,r1,o4c4r4
o3g4r4
7020 DATA o7c4o6g4r4g8e8,r4o5d32
,e32.f32.g32.a32.b32.o6c4r4,o4c4o
3g4o4c4r4
7030 DATA g4g8e8g4g8f8,r1,c4r4o3
g4r4
7040 DATA b4r2g8f8,r4o4a32.b32.o
5c32.d32.e32.f32.g4r4,o4d4o3g4o4d
4r4
7050 DATA g4g8f8g4g8f8,r1,d4o3r4
g4r4
7060 DATA o7d4o6f4r4d8e8,r4o4a32
,b32,o5c32.d32.e32.f32.g4r4,o4d4o
3g4o4d4r4
7070 DATA f4g8f8e4d8f8e4r2,r1r4o
5d32.e32.f32.g32.a32.b32.o6c4,d4r
4o3g4r4o4c4o3g4o4c4
7080 DATA -1
```

MANEJANDO LAS PANTALLAS

¿Cómo y por qué podemos visualizar las letras y símbolos en la pantalla? Un tema muy interesante, pero que muy pocos conocemos. En esta nota hablaremos de la estructura de la parte de video de las MSX.

Las MSX tienen dos modos para textos. La diferencia entre ambos es la cantidad de caracteres que entran por línea y, al mismo tiempo, la cantidad de líneas por pantalla. En el modo 0, se pueden visualizar 40 caracteres por línea y entran 24 filas (o líneas). En el modo 1 esta cifra se reduce a 24 filas de 32 caracteres cada una.

Al encender la computadora, el sistema se pone automáticamente en el modo 0 y no podremos escribir más de 37 caracteres por fila. Esto se debe sólo a efectos de estética de la presentación. Para entender esto, tipiemos la instrucción WIDTH 40 presionando luego RETURN. Hemos seleccionado así, 40 caracteres por línea. Pero notaremos que los caracteres ubicados en los márgenes, no se ven íntegramente. Por eso los diseñadores de este sistema MSX, decidieron darle a la presentación menos cantidad de caracteres por línea para que fueran legibles.

O sea, al encender la máquina, el modo de pantalla presente es "0", pero con 37 caracteres por fila en vez de 40.

Muy bien, esta fue una introducción, ahora profundicemos un poco más este tema.

La pantalla (y acá no interesa el modo en que estemos), tiene 192 filas de pixels (horizontales) donde a su vez entran 256 pixels (columnas verticales).

La memoria de video, VRAM, tiene una longitud de 16.384 bytes. Esta memoria es para uso exclusivo del VDP (procesador de video). Es decir, que a esta memoria no puede acceder la CPU; sólo puede hacer modificar su contenido el VDP.

La distribución de esta memoria depende para cada modo de pantalla.

Pero básicamente, para todos los modos, las áreas de VRAM son:

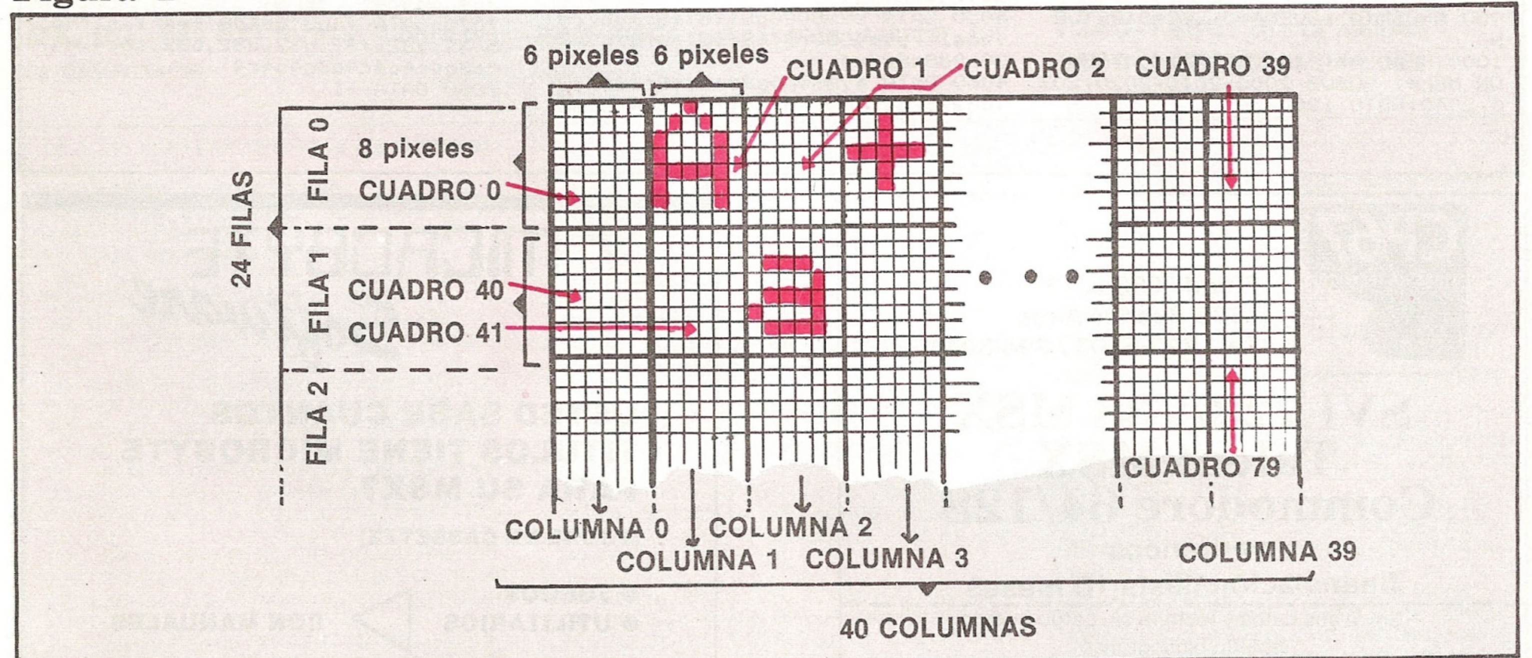
* **tabla de nombres:** indica qué imagen debe aparecer en cierta parte de la pantalla.
* **tabla de generación de patrones:** es el área donde se encuentra el diseño de los caracteres a colocar en la pantalla. A cada uno de estos caracteres definidos les corresponde un valor en la tabla nombres.

Por ejemplo, en la tabla de patrones se encuentra detallado el diseño correspondiente a la letra "a". Y al mismo tiempo, esta letra tiene un número que la identifica en la tabla de nombres.

* **tabla de colores (sólo para los modos 1, 2 y 3):** informa al VDP los colores a usar para el modo que estamos trabajando.
* **tabla de atributos de sprites (para los modos 1, 2 y 3):** guarda las posiciones, el número de patrón y el color para cada sprite.
* **tabla de generación de patrones para sprites (modos 1, 2 y 3):** contiene las definiciones de los sprites, o sea, el diseño de cada uno.

Cada área de la VRAM, es diferente según el modo. No es lo mismo la tabla de generación de patrones en el modo 0 que en el 3.

Figura 1



En este número sólo abarcaremos la memoria de video para los modos 0 y 1. En números siguientes, hablaremos también de los restantes dos.

VRAM PARA EL MODO 0

Para este modo, los 192x256 pixels en que se divide la pantalla se agrupan en cuadros de 8 filas por 6 columnas como se puede ver en el esquema de la figura 1.

Se forman así un total de 24 filas (192/8) con 40 columnas (256/6) de estos cuadrados.

Contamos finalmente con un total de 960 cuadros (24*40=960). En cada cuadro cabe uno solo de los caracteres disponibles como por ejemplo, una letra "a", o un signo "+" o cualquier otro símbolo del teclado o ASCII.

Para trabajar directamente sobre la VRAM, es necesario conocer el comienzo de cada tabla. Esto se consigue a través de la instrucción BASE.

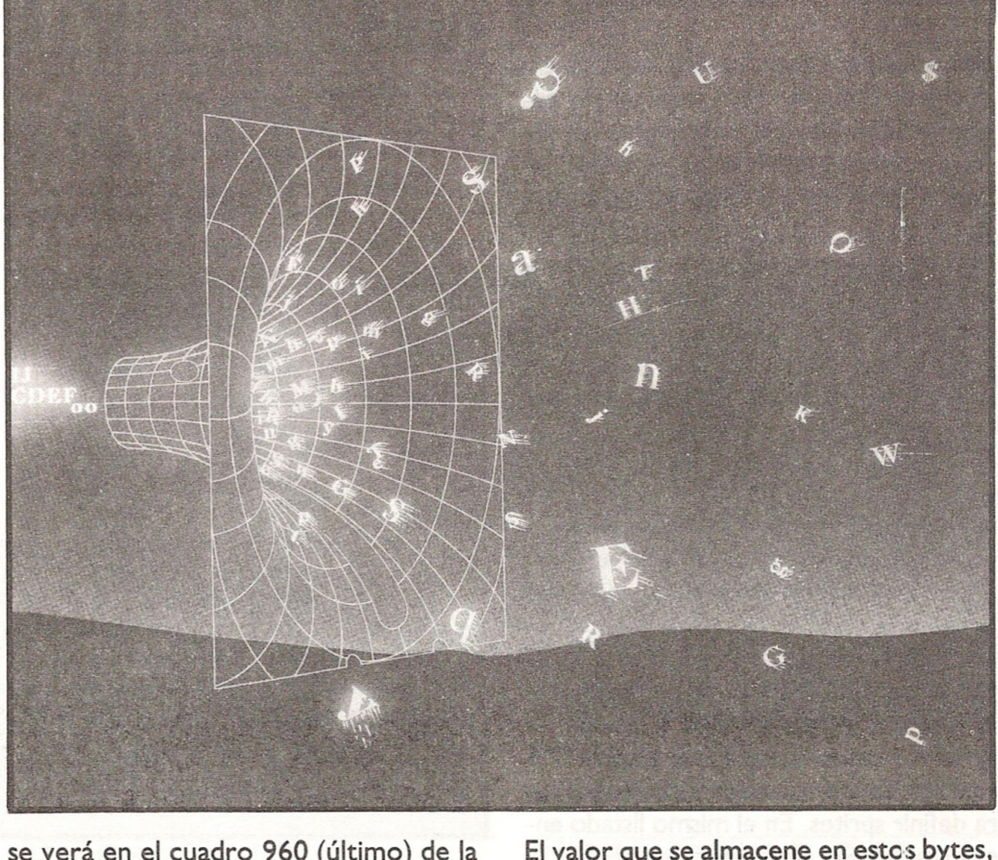


TABLA DE NOMBRES

La sentencia BASE (O) nos dará la dirección donde comienza la tabla de nombres para este modo. Y la sentencia BASE (2) da la dirección donde comienza la tabla de patrones.

Pero veamos un poco cómo se relaciona la tabla de nombres con la pantalla. El byte 0 de la tabla de nombres definirá qué imagen se debe visualizar en el cuadro o casillero 0 de la pantalla. Si en este byte se encuentra almacenado el valor 65, significa que en el casillero 0 de la pantalla se visualizará el caracter de código 65.

El byte 30 tiene la información de qué imagen se debe ver en el casillero 30 y siguiendo así, el último byte de la tabla de nombres (960) indicará qué imagen se verá en el cuadro 960 (último) de la pantalla.

La relación entre la pantalla y esta tabla, la hemos graficado en la figura 2. En la figura 3 está el listado de aplicación de la explicación anterior. Te aconsejamos que antes de seguir leyendo, pruebes tipearlo y ejecutarlo. Te ayudará a entender lo que viene a continuación.

El byte 0 de la tabla de nombres corresponde a la fila 0, columnna 0 de la pantalla; el byte 1 es el de la fila 0, columna 1 y así sucesivamente. Entonces el byte 83 es la fila 2, columna 3 de la pantalla. Esto nos lleva a concluir la siguiente fórmula:

> Número de byte en la tabla
> = Número de línea * 40
> + Número de columna

El valor que se almacene en estos bytes, indicará qué caracter se debe imprimir. Por ejemplo, si el byte 118 tiene el valor 67, indicará que en la fila 2, columna 38 (2*40 + 38= 118), aparecerá el caracter cuyo código ASCII es 67, o sea la letra C.

Figura 2

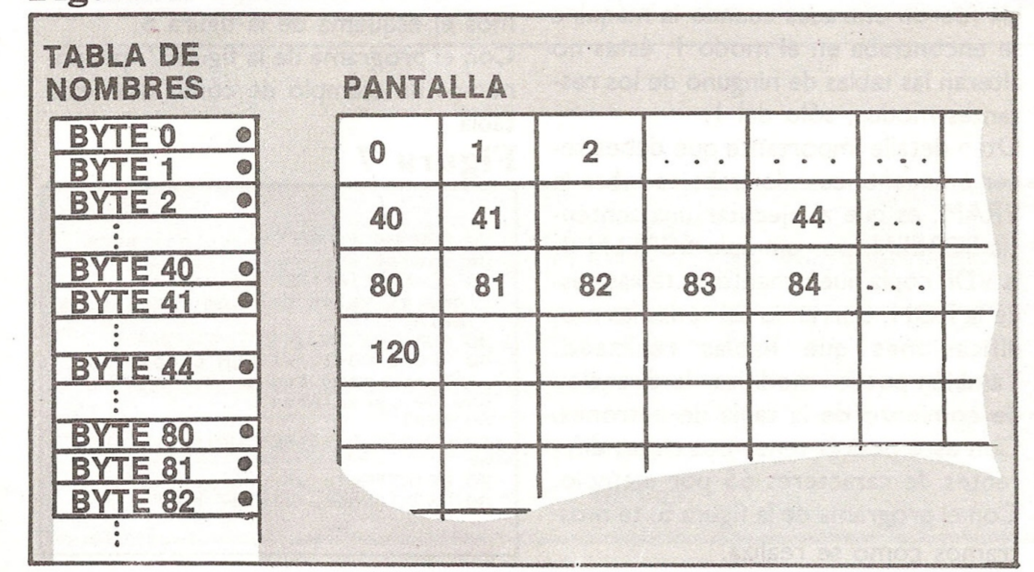


TABLA DE PATRONES O DISEÑOS

Veamos un poco el uso de la tabla de

Figura 3

```
10 SCREEN 0: REM selecciona modo
 de pantalla 0
15 KEY OFF: REM sacamos la impre
sión en pantalla de las teclas de
 función simplemente por estética
.
20 A=BASE(0): REM A tomará el va
lor del comienzo de la tabla de n
ombres, o sea, 0
30 VPOKE A,77: REM colocará en e
l primer casilero de la pantalla,
 el caracter cuyo código ASCII se
a 77. Es decir la letra M.
40 A=A+40: REM ahora A valdrá 0+
40=40
50 VPOKE A,83: REM escribimos en
 el lugar 40 de la pantalla (prim
era columna de la segunda fila),
el caracter 83 que sería una S.
60 A=A+40: REM incrementa el val
or de A y sera: 40+40=80
70 VPOKE A,88: REM colocaremos e
n el cuadro número 80 (primera co
lumna de la tercera fila), la let
ra X que correspondería al código
 88.
80 LOCATE 0,20:PRINT"Para cortar
 el programa presina CTRL/STOP":
REM mensaje final aclarator o
90 GOTO 90 :REM colocamos esta l
ínea para evitar que apaezca el c
ursor al terminar la ejecución de
l programa.
```

patrones para este modo. Aquí se encuentran definidos cada caracter del código ASCII. Lo más interesante es que los podemos modificar; ya veremos cómo. Cada caracter está definido en 8 bytes de la misma forma que definimos sprites. Por esto, la longitud de esta tabla para definir todos los caracteres (en total 256) es de 2K bytes aproximadamente para el modo 0. Observemos que los 2K se deducen de esta cuenta:

256*8 = 2.048.

El primer caracter de ASCII está definido en los primeros 8 bytes (del 0 al 7) de la tabla de patrones. El segundo caracter se encuentra definido entre los bytes 8 y 15.

Intentemos directamente modificar estos caracteres tipeando el programa de la figura 4. Notemos que la forma de redefinir los caracteres no es más que realizar las mismas operaciones que para definir sprites. En el mismo listado encontraremos la explicación de la operación que ejecuta cada línea. Seguramente este programa te ayudará a aclarar la forma de manejar la tabla de diseño; cabe sólo experimentar. Para volver a obtener las definiciones comunes de los caracteres, puedes resetear la computadora o simplemente cambiar de modo de pantalla.

En este programa redefinimos la letra A cuyo caracter ASCII es 65. Esta se encuentra definida entre las posiciones 520 + comienzo de la tabla, (8*65 = 520), hasta la dirección 527 + comienzo de la tabla. (527 = 8*65 + 7). Por eso, el caracter cuyo código ASCII es XX (número ficticio), estará definido entre las posiciones:

BASE(2) + (XX * 8) hasta
BASE(2) + ((XX*8) + 7)

Probemos con distintos valores para los dos programitas (el de la figura 3 y 4), donde usamos estas dos tablas, y obtendremos interesantes resultados.

En este modo no hay tabla de colores, porque el color a usar de tinta y de fondo, se encuentra almacenado en el registro 7 del VDP.

Tampoco hay tabla para sprites, pues este modo no permite usarlos.

No olvidemos que al modificar las tablas de patrones, lo haces para el modo donde te encuentres trabajando.

Si las instrucciones para modificar la tabla fueron entradas cuando la máquina se encontraba en el modo 1, éstas no alteran las tablas de ninguno de los restantes modos, sólo del 1.

Otro detalle importante que debes tener en cuenta cuando trabajes sobre la VRAM, es que al ejecutar una sentencia SCREEN, por ejemplo SCREEN 0, la VDP copia nuevamente las tablas desde la ROM, borrando así todas las modificaciones que habías realizado.

También puedes modificar la dirección de comienzo de la tabla de patrones. Con esto podrás tener dos clases diferentes de caracteres 65 por ejemplo. Con el programa de la figura 5, te mostramos cómo se realiza.

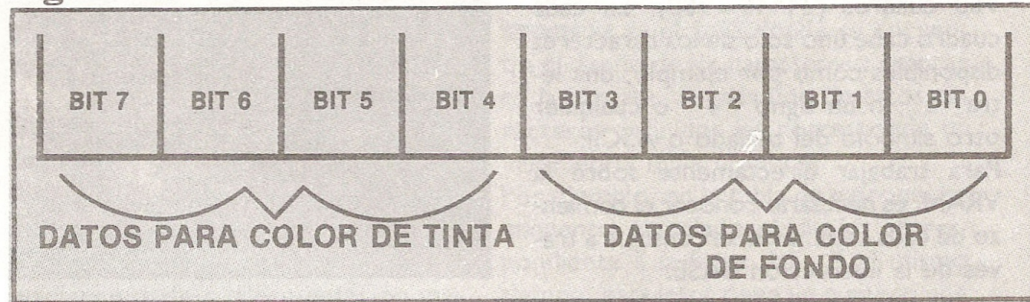
Figura 4

```
10 SCREEN 0: REM selecciona modo
 de pantalla 0
20 KEY OFF:REM sacamos la defini
ción de las teclas de función por
 estética
30 VPOKE 98,65: REM coloca en la
 columna 18 de la tercer fila, la
 letra A.
40 A=BASE(2): REM la variable A
tomará el valor del inicio de la
tabla de diseños para el modo 0.
50 FOR H=1 TO 500:NEXT:REM hacem
os un retardo para ver la letra a
ntes de cambiarla.
60 F=BASE(2)+8*65: REM buscamos
dónde se encuentra la definición
de la letra A.
70 FOR I=F TO F+7 : VPOKE I,74:N
EXT:REM a cada uno de los 8 bytes
 que definen la letra A, los modi
ficamos por la definición 74 como
 lo haciámos con los srites.
80 VPOKE 178,65:REM colocamos el
 nuevo caracter definido, debajo
de la A imprimida antes, pero dej
ando 2 filas libres.
90 LOCATE 0,20:PRINT"Para cortar
 el programa, presioná CTRL/STOP"
100 GOTO 100:REM evitamos con es
to que aparezca el cursor.
```

Figura 5

```
10 SCREEN 0
20 FOR I=0 TO 15: REM Muestra qu
e pasa al inicializar la tabla de
 nombres en distintas direcciones
 -desde 0 a 15-
30 VDP(2)=I:REM Cambia el comien
zo de la tabla de nombres para el
 modo 0
40 VPOKE 450,77:REM Coloca la 'M
'
50 VPOKE 451,83:REM Coloca la 'S
'
60 VPOKE 452,88:REM Coloca la 'X
'
70 FOR F=1 TO 700:NEXT:REM Retar
do para leer cartel
80 NEXT I: REM vuelve a colocar
cartel pero con otro comienzo
90 GOTO 20:REM Vuelve a comenzar
```

Figura 6



MODO 1

Para este modo tenemos más tablas: tabla de nombres cuya dirección de comienzo está dada por BASE (5), tabla de patrones dada por BASE (7), tabla de colores que comienza en la dirección BASE (6), tabla de atributos de sprites en BASE (8) y por último tabla de patrones de sprites, desde la dirección BASE (9).

Las tablas de nombres y patrones se utilizan de la misma forma que en el modo 0, con la diferencia de que la longitud de la tabla de nombres es de 768 bytes. La definición de los caracteres se realiza también de la misma forma que para el modo 0, pero no te olvides que esta tabla comienza en la dirección dada por BASE (7).

TABLA DE COLORES

Veamos cómo se usa la tabla de colores. Esta tiene 32 bytes de largo. El primer byte de la tabla tiene los colores de tinta y de fondo, a usar para los primeros 8 bytes definidos en la tabla de patrones. O sea, son los colores para los caracteres cuyo código ASCII va desde el 0 al 7.

El segundo byte de la tabla de colores indica cuáles son los tonos para los caracteres 8 al 15 y así sucesivamente.

Los bits 7 al 4 tienen el color de tinta y los bits 3 al 0 el de fondo a usar para los 8 caracteres correspondientes. Veamos el esquema de la figura 6.

Con el programa de la figura 7, proponemos un ejemplo de cómo usar esta tabla.

Figura 7

```
10 SCREEN 1:REM Selecciona modo
de pantalla
20 CO=BASE(6):REM La variable CO
 toma el valor del comienzo de la
 tabla.
30 FOR I=1 TO 14
40 VPOKE CO+I,I+1:REM Cambia los
 colores para los primeros 14 gru
pos de caracteres
50 NEXT
60 LOCATE 0,10:REM Ubica el curs
or
70 PRINT"Esto en una prueba"
80 PRINT"Queda claro, no?"
```



CONTROL DE GASTOS

CLASE: COMERCIAL

Con este programa podremos llevar un control de los gastos de un hogar, pero se puede modificar según las necesidades de cada usuario.

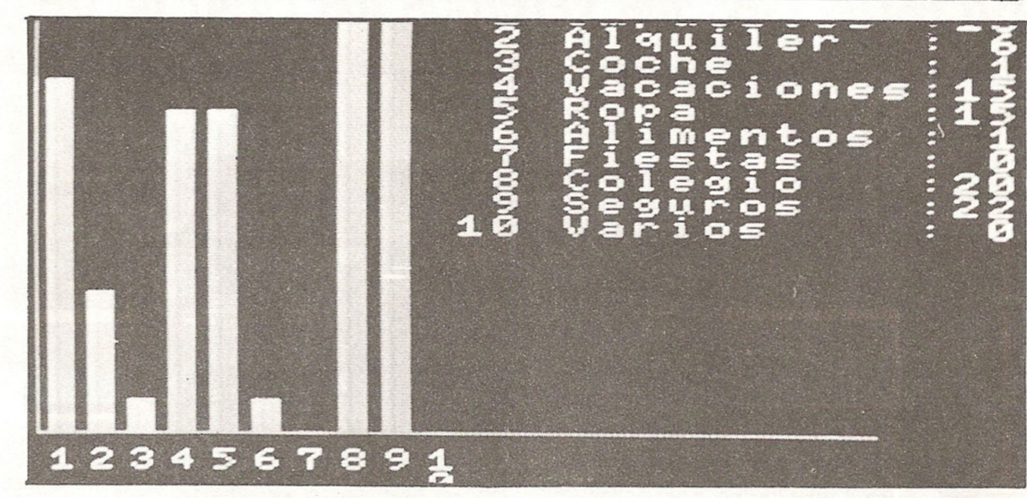
Nos dirá cuáles fueron los gastos efectuados y podremos justificarnos así, porque llegaremos a fin de mes con poco dinero.

Sólo se consideran gastos como impuestos, ropa, seguros, alimentos, etcétera. Pero dada la sencillez con que se realizó este programa, permite ser fácilmente modificado para satisfacer las necesidades de los diferentes usuarios.

Se trata de un software donde daremos entrada a los gastos, detallando el importe y el ítem al que responde.

Obtendremos una representación gráfica sobre el porcentaje de cada gasto, en un sistema de barras.

Gracias a la escasa longitud de este listado, es ideal para emplearlo como subrutina en una agenda (como la publicada en nuestro número 2). De este modo podremos tener control de gastos y de citas en un solo programa.



VARIABLES IMPORTANTES:

N$: matriz con los ítems de los gastos
R$: gasto ingresado
TD: valor del gasto total ingresado
D: matriz con el gasto para cada ítem
DT: gasto total de los ítems
P: porcentaje de cada ítem

ESTRUCTURA DEL PROGRAMA:

10-60: inicializa variables
70-280: entrada de datos y realización de operaciones
290-430: gráfico de barras

```
10 DATA Impuestos,Alquiler,Coche
,Vacaciones,Ropa,Alimentos
20 DATA Fiestas,Colegio,Seguros,
Varios
30 FOR I=1 TO 10
40 READ N$(I)
50 NEXT I
60 OPEN "grp:" FOR OUTPUT AS#1
70 SCREEN 0:KEY OFF:COLOR 4,1,1
80 R$="":INPUT"Total de gastos "
;R$

90 IF VAL(R$)<>0 THEN TD=VAL(R$)
100 IF TD=0 THEN 70
110 MX=0:DT=0
120 PRINT:PRINT:PRINT
130 FOR I=1 TO 9
140 PRINT N$(I);
150 R$="":INPUT R$
160 IF VAL(R$)<>0 THEN D(I)=VAL(
R$)
170 IF D(I)<MX THEN 190
180 MX=D(I)
190 DT=DT+D(I)
200 IF DT<= TD THEN 230
210 PRINT"Total> Gastos"
220 TD=DT
230 NEXT I
240 D(10)=TD-DT
250 IF D(10)>MX THEN MX=D(10)
260 FOR I=1 TO 10
270 P(I)=INT((D(I)/(MX/100)))*1.
6
280 NEXT I
290 SCREEN 2:DRAW"bm17,5":PRINT#
1,"^";TAB(15)"                 Ga
stos    %"
300 LINE(19,7)-(19,170):LINE-(21
4,170)
310 FOR I=1 TO 10
320 COLOR 6
330 FOR J=170 TO 170-P(I) STEP -
1
340 LINE(12+I*10,J)-(18+I*10,J)
350 NEXT J
360 COLOR 15
370 DRAW"bm120,"+STR$(8*I+16):PR
INT#1,USING"##-";I;
380 PRINT#1,N$(I):DRAW"bm224,"+S
TR$(8*I+16):PRINT#1,USING":##";IN
T(D(I)/(TD/100))
390 DRAW"bm"+STR$(13+I*10)+",176
":PRINT#1,MID$(STR$(I),2,1)
400 DRAW"bm"+STR$(13+I*10)+",184
":PRINT#1,MID$(STR$(I)+" ",3)

410 NEXT I
420 A$=INPUT$(1)
430 GOTO 70
```

TOSHIBA HX-20, EN DETALLE

En estas páginas veremos los cuadros de las especificaciones técnicas de esta nueva máquina.





Para comenzar vemos en la figura 1 el cuadro comparativo de la HX-20 y la HX-22. Esta última, hermana de la anterior, todavía no se encuentra en el país.

En la figura 2, vemos la tabla con las nuevas instrucciones del basic extendido de la fabulosa HX-20. Como podemos observar —si es que tenemos alguna noción previa sobre el manejo de instrucciones del drive— éstas, salvo excepciones, son muy similares y de corriente y fácil utilización.

GLOSARIO:

CPU: (Central Procesor Unit), Unidad Central de Proceso o microprocesador.
ROM: Memoria de lectura.
RAM: Memoria de lectura y escritura.
VDP: (Video Display Unit) Chip de video.
PSG: (Program Sound Generator) Chip de sonido.
RF: Radio Frecuencia, salida para Televisor Común.
RS 232C: conector de comunicaciones estandarizadas, que cumple con la norma del mismo nombre.
RGB: (Red Green Blue) salida para monitor con el mismo tipo de entrada.
CA: Corriente Alterna, es el tipo de corriente común de la red domiciliaria.

Instrucciones extendidas	(usadas en combinación con la instrucción CALL)
MEMINI	Inicializa la función RAM de disco.
MEMOFF	Anula la función RAM de disco.
MFILES	Visualiza los nombres de fichero RAM.
MKILL	Borra un fichero RAM.
MNAME	Redenomina un fichero RAM
Comandos	
SAVE	Alamacena programas en un fichero RAM.
LOAD	Carga programas de un fichero RAM a la memoria principal (Area del usuario).
RUN	Ejecuta los programas cargados desde un fichero RAM a la memoria principal.
MERGE	Combina programas de un fichero RAM con programas actualmente en la memoria principal.
Instrucciones	
OPEN	Abre un fichero RAM.
CLOSE	Cierra un fichero RAM.
PRINT#	Envía números y cadenas de caracteres a un fichero RAM.
PRINT#USING	Envía números y cadenas de caracteres a un fichero RAM en un formato especificado.
INPUT#	Introduce números y cadenas de caracteres desde un fichero RAM y los asigna a variables.
LINE INPUT#	Introduce cadenas de caracteres desde un fichero RAM y las asigna a variables de cadena.
Funciones	
INPUT$	Lee en cadena de caracteres de una longitud especificada desde un fichero RAM.
EOF	Verifica el final de un fichero RAM.
LOC	Verifica la posición actual en un fichero RAM.
LOF	Verifica la longitud del fichero RAM.

Mientras que los comandos, instrucciones y funciones usados en el BASIC extendido son los mismos que los mencionados en el MSX BASIC o MSX DISK BASIC, sus formatos o especificaciones están modificados en el BASIC extendido para que se pueda acceder a los ficheros RAM.

usuaria '87

V Congreso Nacional de Informática, Teleinformática y Telecomunicaciones.

Informática y Comunicaciones: Recursos para la excelencia.

Del 1° al 5 de Junio de 1987. Plaza Hotel.

En el marco de Usuaria '87 se llevará a cabo Unimática '87: Primer Encuentro de Integración entre la Universidad y la Empresa.

Presentación de trabajos:
Los resúmenes de los trabajos a presentar deberán ser remitidos antes del 15-12-86 a Usuaria.

Areas de Interés (No Excluyentes)

1. Gobierno
2. Educación
3. Banca
4. Producción
5. Derecho
6. Cultura y Sociedad
7. Inteligencia artificial
8. América Latina
9. Tecnologías informáticas
10. Tecnologías de telecomunicaciones
11. Pequeña y mediana empresa

Organiza **usuaria**

Asociación Argentina de Usuarios de la Informática y las Comunicaciones.

Rincón 326 (1081) Capital Federal.
T.E. 47-2631/2855

ESPECIFICACIONES DE LOS HX-20 y HX-22

ITEM		HX-20	HX-22
CPU		Z80A	
Memoria	ROM	64 KB 32 KB (MSX BASIC) 32 KB (BASIC Extendido + Tratamiento de Textos)	
	RAM	80 KB	
Lenguaje de programación		MSX BASIC	
		FUNCTION RAM DE DISCOS DEL MSX BASIC	BASIC Extendido exclusivo de TOSHIBA para el RS232C
Salida de imagen	VDP	T6950 (Software equivalente al TMS-9918A)	TMS-9929A equivalente
	Tipos de pantalla	Imagen patrón de caracteres 1 pantalla Salida sprite (forma) 32 pantallas Margen 1	
	Salida de imagen patrón de caracteres	4 modos	
	Texto	32 caracteres × 24 líneas (29 diagonales) 40 caracteres × 24 líneas (37 diagonales)	
	Gráfico de alta definición	256 × 192 puntos	
	Multicolor	64 × 48 bloques (1 bloque, 4 × 4 puntos)	
	Indicación sprite	4 tipos de imagen patrón 8 × 8 puntos (estándar), 8 × 8 puntos (ampliada) 16 × 16 puntos (estándar), 16 × 16 puntos (ampliada)	
	Color	16 colores	
Salida de sonido	PSG	AY-3-8910 equivalente	
	Octavas	8 octavas	
	Acordes	Triple acorde	
Teclado	Estructura/Número de teclas	Estructura de pasos cilíndrica/73 teclas	
	Tipos de letras	alfanuméricas, caracteres gráficos	

ITEM		HX-20	HX-22
Interfaces incorporados	Cassette de audio	Método FSK (1200, 2400 baud)	
	Impresora	Especificaciones Centronics	
	Comunicaciones	Ninguna	Especificaciones del RS232C
Conectores exteriores	Zócalo del cartucho	2 (delantero, posterior)	
	Conector de RF	1	
	Conector de video (compuesto)	1	
	Conector de RGB	Ninguno	1
	Conector de audio	1	
	Conector del cassette	1	
	Conector de mando para juegos	2	
	Conector de la impresora	1	
	Conector del RS232C	Ninguno	1
Temperatura y humedad		5-35°C 20-80%	
Alimentación		CA de 220V 50 Hz	
Consumo		12W	19W
Dimensiones (AxAxP)		420x220x75 mm	
Peso		2,7 kg	2,8 kg



MOVIMIENTO DE SPRITES

Si bien los sprites se pueden mover por la pantalla con la instrucción de PUT SPRITE, no se pueden hacer girar o rotar con esta sentencia. Sería muy útil contar con esta ventaja, especialmente para los sprites de los juegos de acción.

En la figura 2 proponemos una imagen sprite, pero que puede ser redefinida con una propia. Nosotros escogimos que este sprite sea de 16x16 ampliado. Pero puede ser de otro tamaño. Si modificamos esta opción, en la línea 10 de la figura 2, deberemos cambiar los parámetros de la sentencia SCREEN.

El listado de la figura 1, corresponde a una pequeña rutina, escrita en lenguaje de máquina.

Figura 1

```
10 SCREEN 1,3
20 FOR Y=0 TO 3
30 FOR X=0 TO 7
40 A$(Y)=A$(Y)+CHR$(VPEEK(X+(Y+4
8)*8))
50 NEXT X,Y
60 FOR X=0 TO 3
70 A$=A$+A$(X)
80 NEXT
90 SPRITE$(0)=A$
100 PUT SPRITE 0,(150,120)
110 Z$=INKEY$:IF Z$="" THEN 110
120 DEF USR7=40000!:A=USR7(NU)
130 GOTO 110
```

Figura 2

```
10 FOR X=40000! TO 40075!:READ V
$
20 POKE X,VAL("&H"+V$):S=S+PEEK(
X)
30 NEXT
40 IF S<>8543 THEN BEEP:CLS:PRIN
T"hay un error"
50 DATA cd,8a,2f,3a,f8,f7,cd,84,
00,3a,e0,f3,e6,02,28,28,e5,eb,21,
10,00,45,19,eb,cd,4a,00,4f,eb,cd,
4a,00,f5,79,cd,4d,00,eb,f1,cd,4d,
00,23,13,10,e9,e1,1e,04,cd,78,9c,
1d,20,fa,c9,16,08,cd,4a,00,06,08,
1f,cb,11,10,fb,79,cd,4d,00,23,20,
ef,c9
```

Esta rutina cumple la función de invertir una sprite en forma vertical, provocando un efecto similar a cuando nos miramos en un espejo.

En la figura 3, mostramos el resultado de la rutina.

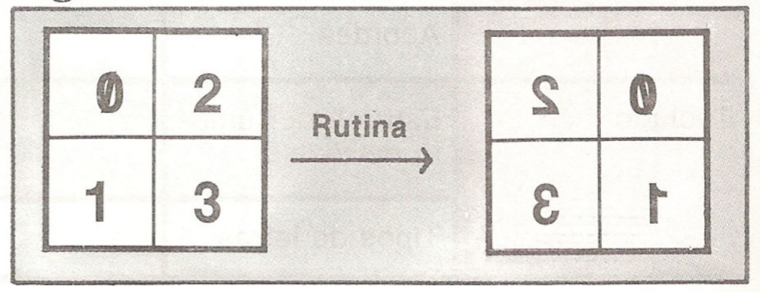
Primero debemos copiar el listado de la primera figura, ejecutarlo, y luego el de la figura 2.

Observemos, que al hacer ejecutar el primer listado, no advertimos ningún cambio.

Sucede que la ejecución de este listado, solo carga en memoria los códigos Assembler, pero sin hacerlo correr. En cambio la figura 2, define un sprite para mover, y en la línea 120 se llama a la rutina de inversión.

Entonces, si queremos incorporar esta rutina en nuestros programas, solo deberemos entrar el listado 1 (como subrutina) y la sentencia: DEF USR7=4000!: A=USR7(NU), donde NU es el número del sprite a mover. En nuestro ejemplo se mantiene en cero.

Figura 3

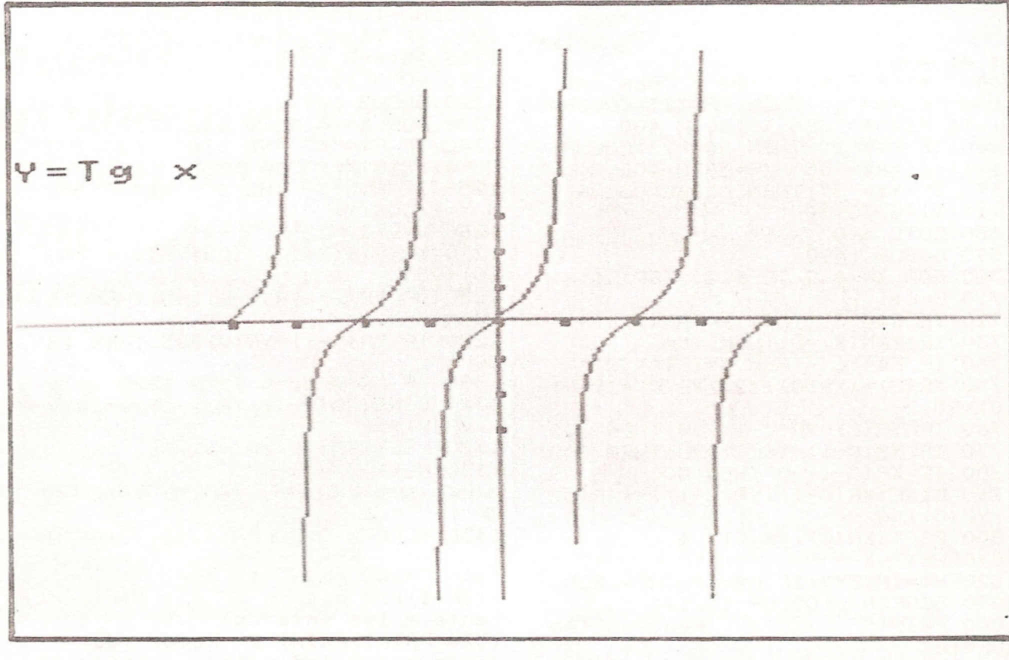




REPASANDO TRIGONOMETRIA

CLASE: EDUCATIVO

Las funciones trigonométricas suelen estudiarse en los últimos años de la secundaria. Para los que han llegado a esta etapa, les damos una mano para repasar este tema.
Algunos profesores escriben las fórmulas que definen a estas funciones trigonométricas y luego nos dan una pila de ejercicios para resolver aplicando las "formulitas"
Sin comprender lo que estamos hablando, incorporamos en nuestro archivo neuronal conceptos con pocos fundamentos.
Nosotros no les daremos una clase magistral de funciones circulares, pero en este programa podremos encontrar los datos principales para cada función y su inversa como: seno, coseno, tangente, secante, arcotangente, arcoseno, etc.
Lo más importante es observar las diferencias gráficas entre cada función. Con este software observaremos el gráfico que representa cada una de las funciones.
Y si se quiere profundizar mas aún, encontraremos explicado el dominio, recorrido, periodicidad, monotonía y continuidad de las mismas.
La primera vez que corramos el programa, es conveniente leer, antes que nada, la información general sobre este tema. Para esto, se debe presionar "O" cuando nos encontremos en el menú inicial.
Luego, con sólo seguir las indicaciones aclaradas en este ítem, será fácil manejar el programa y podremos profundizar los conocimientos de trigonometría.



VARIABLES IMPORTANTES:

R$: opción elegida para estudiar
XR: coordenada X de la función a graficar
YR: coordenada Y de la función a graficar

ESTRUCTURA DEL PROGRAMA:

10-20: inicialización del sistema
30-160: pantalla inicial
170-260: música
270-390: estudia respuesta ingresada
400-680: rutina para el seno y el coseno
690-1020: rutina para la tangente y cotangente
1030-1210: rutina para arcotangente
1220-1500: rutina para secante y cosecante
1510-1680: rutina para arcoseno y arcocoseno
1690-1800: grafica ejes cartesianos
1810-1920: ejes cartesianos para función inversa
1930-2220: instrucciones

```
10 COLOR 1,15,15
20 CLS:KEY OFF:DIM XR(1000),YR(1
000)
30 SCREEN 0:CLS:OPEN"grp:" AS#1:
PRINT"         FUNCIONES TRIGONOMET
RICAS"
40 PRINT:PRINT:PRINT"
  -M E N U"
50 PRINT"              -----------"
60 PRINT:PRINT"1- Función seno"
70 PRINT"2- Función coseno""
80 PRINT"3- Función tangente"
90 PRINT:PRINT"4- Función coseca
nte"


100 PRINT"5- Función secante"
110 PRINT"6- Función cotangente"
120 PRINT:PRINT"7- Función arcos
eno"
130 PRINT"8- Función arcocoseno"
140 PRINT"9- Función arcotangent
e"
150 PRINT:PRINT"0- Información"
160 PRINT:PRINT"F- Fin"
170 A$="v15o4s1m1000t100L40"
180 B$="ccfcfafr25ffafa"
190 C$="o5co4ar25fao5co4r25"
200 D$="afcr25ccfr25L30fr35L10f"
210 FOR I=1 TO 1
220 PLAY"xa$;"
230 PLAY"xb$;"
240 PLAY"xc$;"
250 PLAY"xd$;"
260 NEXT I
270 R$=INKEY$
280 IF R$="1" THEN A$="Sen x":GO
TO 400
290 IF R$="2" THEN A$="Cos x":GO
TO 400
300 IF R$="3" THEN A$="Tg x":GOT
O 690
310 IF R$="6" THEN A$="cotg x":G
OTO 690
320 IF R$="4" THEN A$="Cosec x":
GOTO 1220
330 IF R$="5" THEN A$="Sec x":GO
TO 1220
340 IF R$="7" THEN A$="Arc sen x
":GOTO 1510
350 IF R$="8" THEN A$="Arc cos x
":GOTO 1510
360 IF R$="9" THEN A$="Arc tg x"
:GOTO 1030
370 IF R$="0" THEN 1930
380 IF R$="F" OR R$="f" THEN END
390 GOTO 170
400 GOSUB 1690
410 FOR X=-6.2 TO 6.2 STEP .1
420 IF R$="1" THEN Y=SIN(X)
430 IF R$="2" THEN. Y=COS(X)
440 XR=(X*10)+125:YR=(Y*-10)+95
450 PSET(XR,YR),6
460 NEXT X
470 H$=INKEY$:IF H$="" THEN 470
480 SCREEN 0:PRINT"              Fu
nción seno"
490 COLOR 4,10,10:PRINT:PRINT"1-
 Dominio: los reales"
500 PRINT"2- Recorrido: [-1,1].
El seno está  siempre comprendid
o entre el -1 y 1"
510 PRINT"3- Periodicidad: Es pe
riódica y su  período es de 2π.
Se repite indefi-  nidamente el g
ráfico en el intervalo [2π,4π] ";
520 PRINT"los valores que tuvo e
n     [0,2π].":PRINT"Es decir: s
en(x+2π)=sen x"
530 PRINT"4- No es inyectiva ni
suryectiva"
540 PRINT"5- Monotonia: -Es crec
iente en        (-π/2,π/2),(3π/2,5
π/2)..."
550 PRINT"                -Es decr
eciente en      (π/2,3π/2),(5π/2,7
π/2)..."
```

```
560 PRINT"5- Es continua."
570 PRINT:PRINT"Pulsa una tecla"
580 Y$=INKEY$:IF Y$="" THEN 580
590 PRINT"          Función cosen
o"
600 PRINT"Las propiedades son la
s mismas salvo en lo referente a
la monotonía."
610 PRINT"El coseno es creciente
 en los inter- valos (-π,0),(π,2π
)"
620 PRINT"El coseno es decrecien
te en (0,π),(2P,3π)"
630 PRINT:PRINT"1- Seno
                    2- Coseno
                          3- Volve
r al menú"
640 Y$=INKEY$:IF Y$="" THEN 640
650 IF Y$="1" THEN R$="1":COLOR
1,11,11:A$="SEN X":GOTO 400
660 IF Y$="2" THEN R$="2":COLOR
1,11,11:A$="COS X":GOTO 400
670 IF Y$="3" THEN CLOSE:COLOR 1
,11,11:GOTO 30
680 GOTO 640
690 GOSUB 1690
700 FOR X=-6.2 TO 6.2 STEP .1
710 C=C+1
720 IF R$="3" THEN Y=TAN(X)
730 IF TAN(X)=0 THEN 750
740 IF R$="6" THEN Y=1/TAN(X)
750 XR(C)=(X*10)+125:YR(C)=(Y*-1
0)+95
760 IF YR(C)-YR(C-1)>95 THEN 800
770 IF YR(C-1)-YR(C)>95 THEN 800
780 IF XR(C-1)=0 THEN 800
790 LINE(XR(C-1),YR(C-1))-(XR(C)
,YR(C)),6
800 PSET(XR(C),YR(C)),6
810 NEXT X
820 H$=INKEY$:IF H$="" THEN 820
830 SCREEN 0:COLOR 11,1,1
840 PRINT"          Función TAN
GENTE"
850 PRINT:PRINT"1- Dominio:R-> (
(2K+1)π/2           con K per
teneciente a Z)"
860 PRINT"2- Recorrido: los real
es"
870 PRINT"3- Periodicidad: Es pe
riódica y su      período es π"
880 PRINT"4- No es inyectiva.Si
es suryectiva"
890 PRINT"5- Es Creciente"
900 PRINT"6- Es 'discontinua' cu
ando x no       pertenece al do
minio"
910 PRINT:PRINT"          Funció
n COTANGENTE"
920 PRINT:PRINT"1- Dominio:R-> (
Kπ con K          pertenecient
e a Z)"
930 PRINT"2- Es siempre decrecie
nte"
940 PRINT"Las restantes propieda
des son      iguales a las de l
a función Tangente"
950 PRINT:PRINT"Pulsa una tecla"
960 Y$=INKEY$:IF Y$="" THEN 960
970 PRINT:PRINT:PRINT"1- Tangent
e                        2- Cot
angente                       3-
 Volver al menú"
980 Y$=INKEY$:IF Y$="" THEN 980
990 IF Y$="1" THEN R$="3":COLOR
1,11,11:A$="TG X":GOTO 690
1000 IF Y$="2" THEN R$="6":COLOR
 1,11,11:A$="COTG X":GOTO 690
1010 IF Y$="3" THEN CLOSE:C=0:CO
LOR 1,11,11:GOTO 30
1020 GOTO 980
1030 GOSUB 1810
1040 FOR Y=-6.2 TO 6.2 STEP .1
1050 B=B+1
1060 X=TAN(Y)
1070 XR(B)=(X*10)+125:YR(B)=(Y*-
10)+95
1080 IF XR(B-1)-(XR(B))>125 THEN
 1110
1090 IF YR(B-1)=0 THEN 1110
1100 LINE(XR(B-1),YR(B-1))-(XR(B
),YR(B)),6
1110 PSET(XR(B),YR(B)),6
1120 NEXT Y
1130 H$=INKEY$:IF H$="" THEN 113
0
1140 SCREEN 0:COLOR 10,1,1:PRINT
"        Función Arcotangente"
1150 PRINT:PRINT:PRINT"Las funci
ones circulares no son        inye
ctivas. Por eso, sus inversas
no son funciones, sino simples
     correspondencias."
1160 PRINT:PRINT:PRINT"El domini
o de esta correspondencia     es s
on los 'reales'"
1170 PRINT:PRINT:PRINT"1- Arcota
ngente                       2- Vo
lver al menú"
1180 Y$=INKEY$: IF Y$="" THEN 11
80
1190 IF Y$="1" THEN A$="ARC TG X
":COLOR 1,15,15:GOTO 1030
1200 IF Y$="2" THEN CLOSE:COLOR
1,15,15:B=0:GOTO 30
1210 GOTO 1180
1220 GOSUB 1690
1230 FOR X=-6.2 TO 6.2 STEP .1
1240 IF R$="4" AND SIN(X)<>0 THE
N Y=1/SIN(X):GOTO 1260
1250 IF R$="5" AND COS(X)<>0 THE
N Y=1/COS(X)
1260 V=V+1
1270 XR(V)=(X*10)+125:YR(V)=(Y*-
10)+95
1280 IF YR(V)-YR(V-1)>95 THEN 13
20
1290 IF YR(V-1)-YR(V)>95 THEN 13
20
1300 IF XR(V-1)=0 THEN 1320
1310 LINE(XR(V-1),YR(V-1))-(XR(V
),YR(V)),6
1320 PSET(XR(V),YR(V)),6
1330 NEXT X
1340 H$=INKEY$:IF H$="" THEN 134
0
1350 SCREEN 0:COLOR 4,10,10:PRIN
T"         Función Secante"
1360 PRINT:PRINT"1- Dominio: R->
((2K+1)π/2 con K           pertenec
iente a los enteros)"
1370 PRINT:PRINT"2- Recorrido: (
desde menos infinito       hasta el
 -1 inclusive) U (desde 1      hast
a el + infinito)"
1380 PRINT:PRINT"3- Es periódica
 de 2π"

1390 PRINT:PRINT"4- Es discontin
ua cuando X no            pertenec
e al Dominio"
1400 PRINT:PRINT:PRINT"
    Función COSECANTE"
1410 PRINT:PRINT"1- Dominio: R-(
Kπ con K ∈ Z)"
1420 PRINT:PRINT"Las demás propi
edades son como las de y=sec x"
1430 PRINT:PRINT"Pulsa una tecla
"
1440 Y$=INKEY$:IF Y$="" THEN 144
0
1450 PRINT:PRINT:PRINT"1- Secant
e                          2- Co
secante                          3
- Volver al menú"
1460 Y$=INKEY$:IF Y$="" THEN 146
0
1470 IF Y$="1" THEN R$="5":A$="s
ec x":COLOR 1,15,15:GOTO 1220
1480 IF Y$="2" THEN R$="4":A$="c
osec x":COLOR 1,15,15:GOTO 1220
1490 IF Y$="3" THEN CLOSE:V=0:CO
LOR 1,15,15:GOTO 30
1500 GOTO 1460
1510 GOSUB 1810
1520 FOR Y=-6.2 TO 6.2 STEP .1
1530 IF R$="7" THEN X=SIN(Y)
1540 IF R$="8" THEN X=COS(Y)
1550 XR=(X*10)+125:YR=(Y*-10)+95
1560 PSET(XR,YR),6
1570 NEXT Y
1580 H$=INKEY$:IF H$="" THEN 158
0
1590 SCREEN 0:COLOR 15,4,4
1600 PRINT:PRINT:PRINT"      Func
ión Arcoseno-Arcocoseno"
1610 PRINT:PRINT:PRINT"Las funci
ones circulares no son        inye
ctivas. Por eso, sus inversas
no son funciones, sino simples
     correspondencias."
1620 PRINT"Tanto la corres
pondencia y=arc cos X como y=arc
senX tienen como dominio  el inte
rvalo [-1,1]"
1630 PRINT:PRINT:PRINT"1- Arcsen
x                          2- Ar
ccos X                           3
- Volver al menú"
1640 Y$=INKEY$:IF Y$="" THEN 164
0
1650 IF Y$="1" THEN R$="7":A$="a
rcsen x":COLOR 1,15,15:GOTO 1510
1660 IF Y$="2" THEN R$="8":A$="a
rccos x":COLOR 1,15,15:GOTO 1510
1670 IF Y$="3" THEN CLOSE:COLOR
1,15,15:GOTO 30
1680 GOTO 1640
1690 SCREEN 2
1700 PSET(10,40),15:PRINT#1,"Y="
;A$
1710 LINE(0,0)-(251,191),1,B
1720 LINE (125,0)-(125,191),6
1730 LINE (0,95)-(251,95),5
1740 FOR X=61 TO 185 STEP 15.5
1750 PSET(X,93),15:PRINT#1,"."
1760 NEXT X
1770 FOR Y=57 TO 129 STEP 12
1780 PSET(123,Y),15:PRINT#1,"."
1790 NEXT Y
1800 RETURN
1810 SCREEN 2
1820 PSET(10,30),15:PRINT#1,"Y="
;A$
1830 LINE(0,0)-(251,191),1,B
1840 LINE (125,0)-(125,191),6
1850 LINE (0,95)-(251,95),5
1860 FOR X=88 TO 158 STEP 11
1870 PSET(X,92),15:PRINT#1,"."
1880 NEXT X
1890 FOR Y=30 TO 154 STEP 15.5
1900 PSET(123,Y),15:PRINT#1,"."
1910 NEXT Y
1920 RETURN
1930 COLOR 4,10,10:SCREEN 0:PRIN
T"         Funciones trigonométric
as"
1940 PRINT:PRINT:PRINT"Los punto
s de corte con los ejes son:"
1950 PRINT"                   Y"
1960 PRINT"                   █ 3"
1970 PRINT"                   █ 2"
1980 PRINT"                   █ 1"
1990 PRINT"                   █ "
2000 PRINT"LLLLLLLLLLLLL█LLLLLL
LLLLLLLLLLL X "
2010 PRINT"-2π -3π -π -π █0  π
π  3π  2π"
2020 PRINT"       2      2 █-1 2
   2"
2030 PRINT"                   █-2"
2040 PRINT"                   █-3"
2050 PRINT:PRINT"En las funcione
s inversas (arc sen,  arc cos, ar
c tg) las coordenadas son contrar
ias:"
2060 PRINT:PRINT"Eje x=-3 -2 -2
 0  1  2  3"
2070 PRINT"Eje y=-2π -3π/2 -π -π
/2  0"
2080 PRINT"   Pulsa una tecla pa
ra continuar"
2090 T$=INKEY$:IF T$="" THEN 209
0
2100 CLS:PRINT"           Funcion
es trigonométricas"
2110 PRINT"            LLLLLLLLLLL
LLLLLLLLLLLLLLL"
2120 COLOR 11,1,1
2130 PRINT"Al elegir un número c
orrespondiente  a alguna función
trigonométrica, en"
2140 PRINT"pantalla saldrá la re
presentación    gráfica de dicha
función, la cual se mantendrá en
pantalla hasta que se"
2150 PRINT"presione alguna tecla
."
2160 PRINT:PRINT"Al pulsar la te
cla, saldrá en        pantalla un
 estudio de la función     represe
ntada."
2170 PRINT"Después, podrás volve
r a ver la      representación, o
 bien, volver al   menú"
2180 PRINT:PRINT" 1- Volver al m
enú                    2- Repetir
"
2190 V$=INKEY$:IF V$="" THEN 219
0
2200 IF V$="1" THEN CLOSE:COLOR
1,15,15:GOTO 30
2210 IF V$="2" THEN 1930
2220 GOTO 2190
```

DESNUDAMOS EL CHIP DE SONIDO

2da. Parte

Continuamos describiendo al responsable de que nuestra MSX sea capaz de formar envolventes y mezclar sonidos. Así completamos la explicación para entender qué sucede cuando creamos música desde el Basic.



Figura 6

R15 BITS				
B3	B2	B1	B0	
CONTINUA	ATAQUE	ALTERNA	SOSTENIDA	
0	0	x	x	
0	1	x	x	
1	0	0	0	
1	0	0	1	Ver detalle en fig. 7
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

EP

En el número anterior hicimos la presentación formal del chip encargado de producir el sonido en nuestra MSX. Como recordarán, cada patita del chip tiene una función o registro que le indica al semiconductor qué debe hacer. Ahora continuamos describiendo esas funciones.

FORMAR ENVOLVENTES

Los envolventes se generan a través de tres registros. El registro R13 crea la forma del envolvente (ver formas en la figura 6) y los registros R11 y R12 marcan la duración de las rampas.

El registro R13 utiliza los 4 bits de menos peso llamados:

* **HOLD** (mantener): al estar en nivel lógico 1 y si acaba el primer ciclo de variación de la forma envolvente, mantendrá el valor con que finalice la rampa, así sea ascendente o descendente, en cuyos casos mantendrá el volumen máximo o silencio respectivamente.

Pero al ser el nivel lógico 0, se repetirá cíclicamente la envolvente.

* **ALT** (alternancia): alterna las rampas en ascendentes y descendentes en el caso de estar con nivel lógico 1.

* **ATT** (ataque): indica de qué forma debe comenzar la forma envolvente: ascendente si el nivel lógico dado es 1 y descendente para nivel lógico 0.

* **CONT** (continuación): al tener el nivel en 1 indica que la forma de la envolvente será la definida por HOLD, pero si es cero al terminar el primer ciclo, volverá a estar en silencio.

La duración entre la primera y segunda rampa está dada por los registros R12 y R11 de la siguiente forma:

$$T = 256 * \frac{R11 + 256*R12}{fr}$$

donde fr (como antes) es la frecuencia del reloj.

Para afectar a un canal por la envolvente, habrá que colocar el bit 4 de su registro (R8 para A, R9 para B y R10 para C) de control de envolvente a "1". De lo contrario su volumen será el fijado por los bits 0 a 3 de dicho registro que sustituirán a la entrada del convertidor digital/analógico de envolvente a los 4 bits del segundo contador.

Figura 7

VOLTAJE NORMALIZADO

1 v — 707 v. — 5v — 303v — 25v — 1515v — 125v — 0

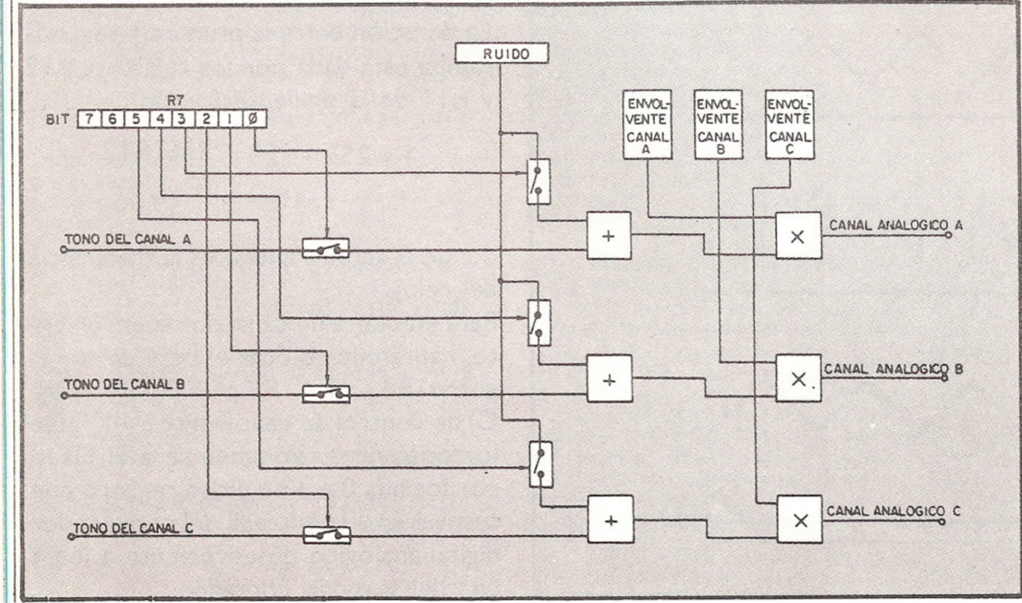
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 | 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

VALOR DECIMAL A LA SALIDA DEL CONTADOR

EP (1/fe) EP (1/fe)

EP, PERIODO

Figura 8



MEZCLA DE SONIDOS

Hemos visto cómo se genera cada sonido, pero faltaría entender qué puede pasar para que escuchemos el sonido final. Vayamos uniendo las piezas de este rompecabezas: tenemos un tono base, al que se le "suma" un ruido y al darle volumen decimos que lo "multiplicamos", y a esto agregamos el ruido por a envolvente de volumen o también puede ser un volumen fijo. Todo esto se realiza dentro del generador de ruido y del generador de envolvente produciendo luego del sonido captado por nuestros oídos.

Esquematizamos un poco más técnicamente esta explicación en la figura 8.

REGISTRO DE CONTROL

El encargado de mezclar los ruidos y tonos que se generan es el registro R7. Está compuesto por tres bloques: "tono", "ruido", ambos de tres bits (uno para cada canal) y "entrada/salida" de dos bits, uno para cada puerto (A y B). Los tres primeros bits de este registro (bit 0-bit 2) controlan el tono a ejecutar por cada canal. Si el nivel lógico es 1, el canal correspondiente impedirá que suene el tono fijado por los registros de control de tono y si es cero el nivel lógico, ejecutará el tono en el canal asignado.

En cambio los bits 3 y 4 (siempre del registro R7) controlan el ruido. Si le damos nivel cero, haremos que se sume el ruido al tono del canal.

Los dos restantes bits (bit 6 y bit 7), fijan la dirección del puerto que se quiere operar. Se usa el bit 6 para el puerto A y el bit 7 para el B.

Al valer alguno de estos bits 1, hará que al llegar un dato del procesador al puerto controlado por el bit correspondiente, este mantendrá el valor del dato en las líneas del bus de datos hasta que sea leído.

En caso que el nivel lógico sea 0, al querer leer, el procesador, el registro del puerto, leerá en realidad un valor binario que se encuentra en las líneas del bus exterior de datos.

Al primer modo se lo llama "salida" y al segundo "de entrada". Tratar de escribir en un puerto de entrada será en vano porque no reconocerá la orden. Cada puerto debe respetar su operación. Finalmente en la figura 9 representamos detalladamente la estructura interna del PSG.

Así hemos visto las funciones de cada patita del chip y podremos entender qué sucede cuando creamos música desde el Basic.

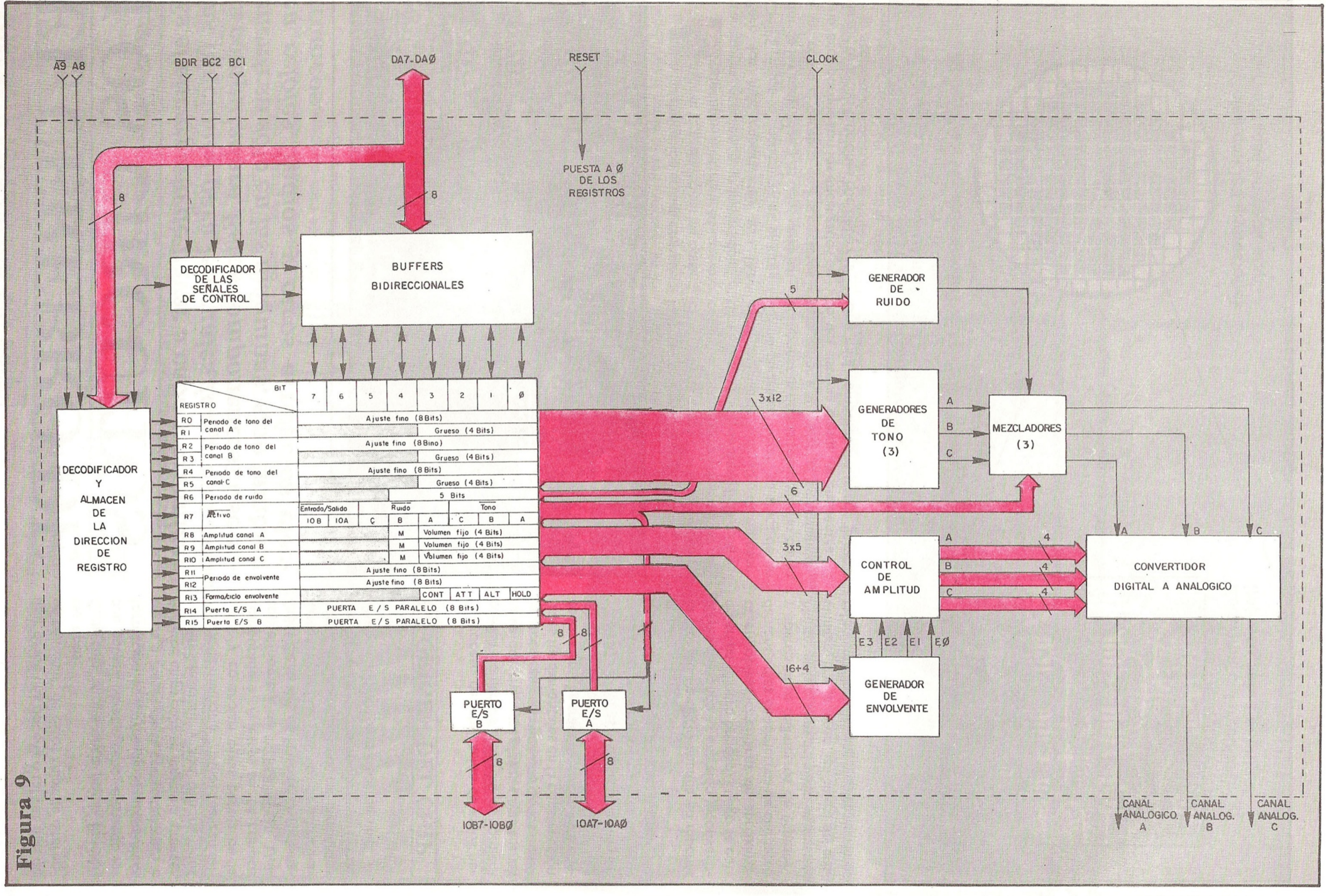
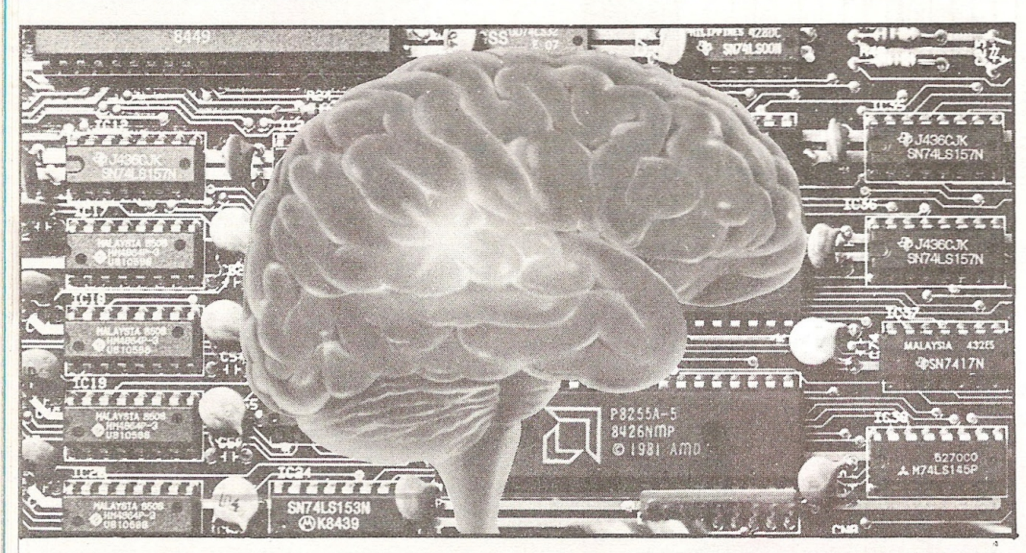

A9 A8
BDIR BC2 BC1
DA7-DAØ
RESET
PUESTA A Ø DE LOS REGISTROS
CLOCK
DECODIFICADOR DE LAS SEÑALES DE CONTROL
BUFFERS BIDIRECCIONALES
DECODIFICADOR Y ALMACEN DE LA DIRECCION DE REGISTRO
REGISTRO
BIT
7 6 5 4 3 2 1 Ø
R0 R1 Periodo de tono del canal A
Ajuste fino (8 Bits)
Grueso (4 Bits)
R2 R3 Periodo de tono del canal B
Ajuste fino (8 Bino)
Grueso (4 Bits)
R4 R5 Periodo de tono del canal C
Ajuste fino (8 Bits)
Grueso (4 Bits)
R6 Periodo de ruido
5 Bits
R7 Activo
Entrada/Salida
Ruido
Tono
IOB IOA C B A C B A
R8 Amplitud canal A
M Volumen fijo (4 Bits)
R9 Amplitud canal B
M Volumen fijo (4 Bits)
R10 Amplitud canal C
M Volumen fijo (4 Bits)
R11 R12 Periodo de envolvente
Ajuste fino (8 Bits)
Ajuste fino (8 Bits)
R13 Forma/ciclo envolvente
CONT ATT ALT HOLD
R14 Puerta E/S A
PUERTA E/S PARALELO (8 Bits)
R15 Puerta E/S B
PUERTA E/S PARALELO (8 Bits)
8
5
3x12
6
3x5
16+4
GENERADOR DE RUIDO
GENERADORES DE TONO (3)
MEZCLADORES (3)
CONTROL DE AMPLITUD
GENERADOR DE ENVOLVENTE
E3 E2 E1 EØ
CONVERTIDOR DIGITAL A ANALOGICO
A B C
4
PUERTO E/S B
PUERTO E/S A
IOB7-IOBØ
IOA7-IOAØ
CANAL ANALOGICO. A
CANAL ANALOG. B
CANAL ANALOG. C

Figura 9

EL CEREBRO DE LOS CIRCUITOS

Es el encargado de llevar a cabo y organizar las funciones para las cuales fue preparada la computadora. Estos componentes no se encuentran sólo en las máquinas, sino también en relojes, radios y otros artefactos electrónicos.



En la fabricación de una computadora intervienen varios componentes electrónicos como transitores, resistencias e integrados. Pero los más importantes son los CHIPS. Estos se encuentran en el interior de los integrados. Los chips vendrían a ser como el cerebro de los circuitos impresos porque son los encargados de dirigir todas las funciones que debe realizar la computadora.

El tamaño de estos cerebros electrónicos es de aproximadamente medio centímetro cuadrado de superficie, con escasas décimas de milímetro de espesor. Como podemos notar, se trata de un componente demasiado pequeño como para tener a su cargo el funcionamiento del sistema donde se encuentra.

Una mini computadora puede llegar a usar desde una docena o tal vez más de un centenar de estos chips de diferentes tipos.

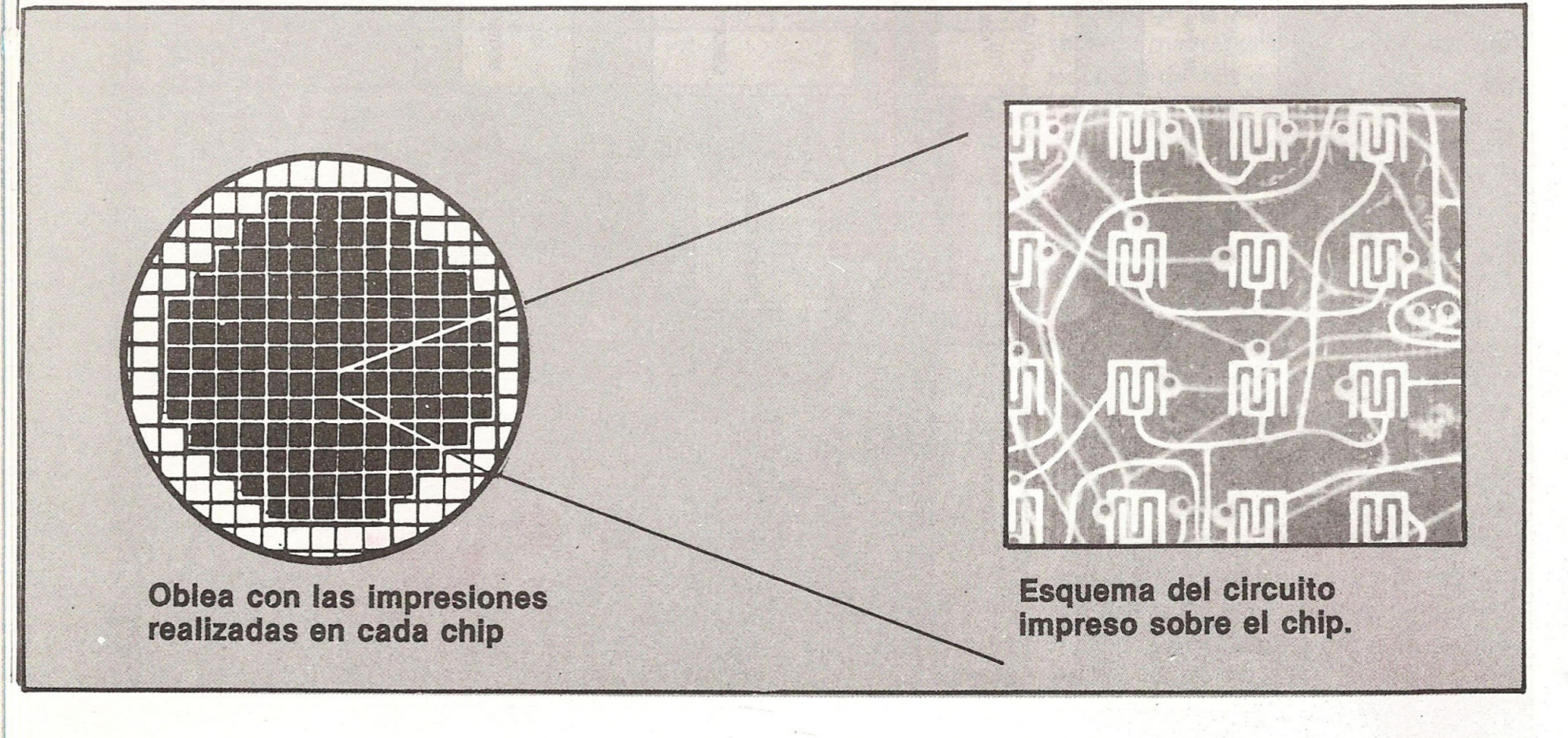
Cada chip se diferencia por la función que deberá llevar a cabo, enviando información en forma de impulsos eléctricos a través de la computadora.

Algunos chips sólo se limitan a almacenar información, y son denominados "chips de memoria", otros realizan cálculos y controlan otras partes de la computadora. A éstos se los conoce con el nombre de "CPU" o sea, Unidades Centrales de Proceso. Afortunadamente para los usuarios de computadoras, el silicio es uno de los elementos más comunes y económicos de la Tierra. Veamos ahora cómo nace un chip.

EL ORIGEN DEL CHIP

El proceso de fabricación de estos pequeños elementos electrónicos comienza en determinadas playas cuyas arenas están compuestas por uno de los materiales fundamentales en la construcción de los chips. Se trata del silicio. Este cristal cuya pureza es de alrededor del 99.9999999 % es la materia prima con la cual son construidos los circuitos centrales de las computadoras, es decir, los chips.

El bióxido de silicio abunda en muchas playas de donde es extraído y convertido en un cilindro brilloso. Luego se corta en obleas circulares de aproximadamente diez centímetros de diámetro y apenas unos milímetros de espesor. Mantener la pureza del silicio es imprescindible, por lo tanto, las obleas del cristal son manipuladas por técnicos vestidos



Oblea con las impresiones realizadas en cada chip

Esquema del circuito impreso sobre el chip.

como cirujanos en vez de ingenieros electrónicos; y en ambientes antisépticos.

Estos técnicos revestidos de guardapolvos, gorras, guantes y máscaras, deben impedir que cualquier partícula de la piel o cabello, que continuamente se desprenden de nosotros, tengan contacto con la oblea de silicio.

FABRICACION DEL CHIP

Las rodajas de silicio puro son la base donde se construirán cientos de chips de pocos milímetros de tamaño.

Primero se diagrama el diseño que llevará el chip.

En el principio de la era de la creación de estos pequeños cerebros, los diagramas eran bastantes sencillos y se realizaban fácilmente sobre papel.

Pero los diseños de los chips eran cada vez más complicados porque se buscaba ampliar sus aplicaciones y se implementaban cada vez más elementos dentro del circuito del chip.

Continuar diagramando entonces el interior de esos componentes sobre papel se hacía más difícil .

Los ingenieros electrónicos tuvieron que recurrir, después de varios estudios y pruebas, a otros métodos que facilitaran la tarea de creación de los circuitos complejos que llevarían finalmente los chips.

Gracias al avance sufrido en todos los aspectos de la electrónica, se inventaron métodos más sofisticados para fabricar circuitos sobre silicio.

También, debido a la crecida demanda de chips, se buscó la forma de fabricar mayor cantidad en el mismo espacio.

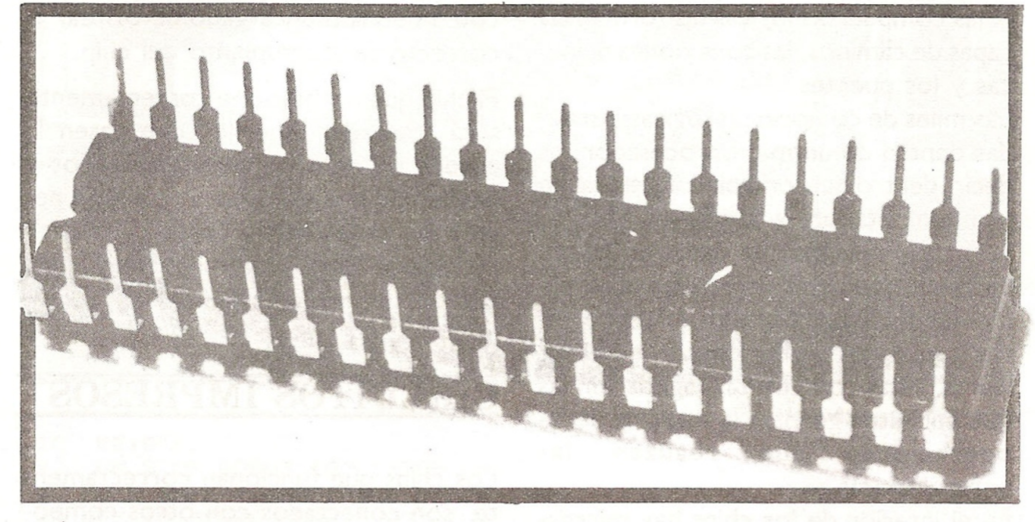
El costo de producción del chip es independiente del número de componentes en su interior. Por lo tanto si un fabricante puede hacer que un chip haga el trabajo de dos, doblando el número de componentes en su interior, el costo de este chip doble será menor que la suma de los dos chips individuales.

Esta es la razón por la cual actualmente las computadoras tienden a bajar sus precios y son más potentes.

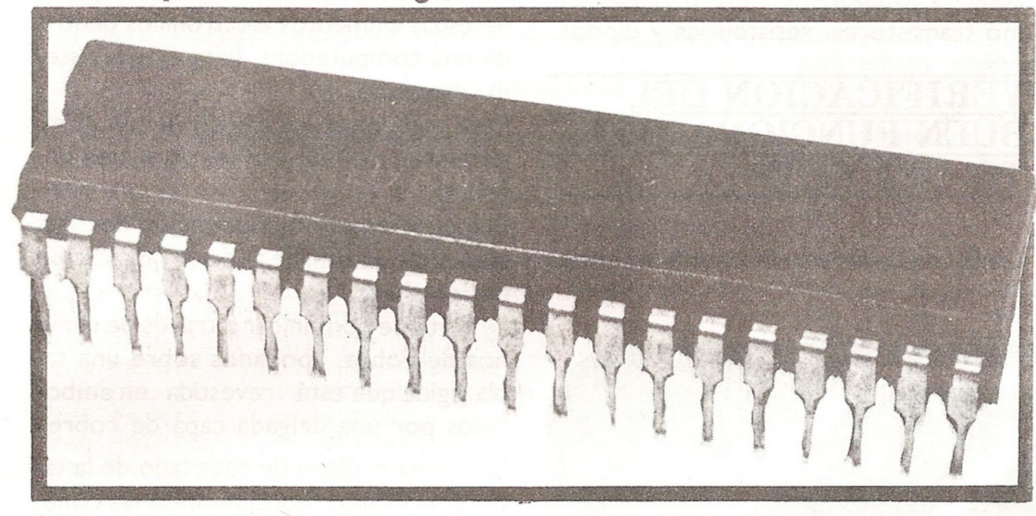
Los circuitos que luego se transformarán en caminos microscópicos sobre el chip son primero dibujados, ampliado 250 veces su real tamaño.

La incesante búsqueda de ampliar las funciones de cada chip ha hecho que su diseño sea demasiado complicado, debiéndose recurrir a la ayuda de las mismas computadoras para diseñar y dibujar los circuitos.

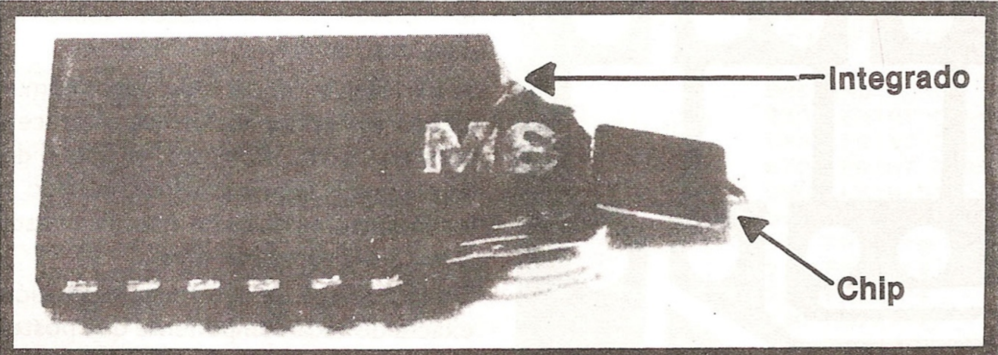
Vista inferior de un integrado



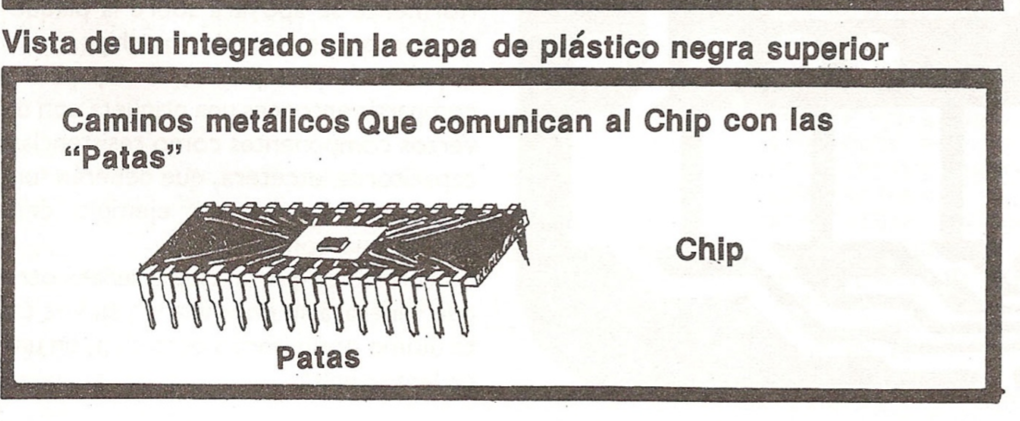
Vista superior de un integrado



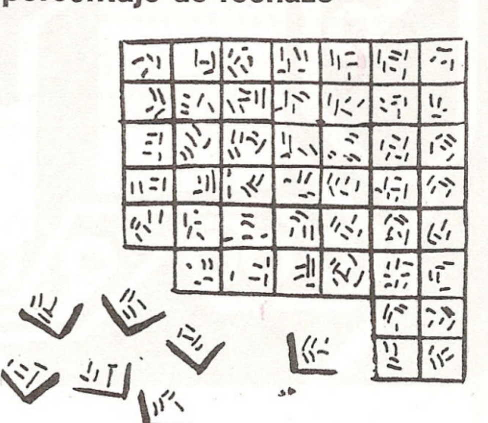
Vista de porción de integrado



Vista de un integrado sin la capa de plástico negra superior



Control de calidad con alto porcentaje de rechazo



De esta manera se entra en un ciclo donde las computadoras facilitan la fabricación de otras futuras, pero con más potencia.

Cuando los planos de los circuitos están terminados, se transfieren a imágenes fotográficas. Un laberinto de caminos intercomunicados, es reducido al tamaño natural del chip, es decir, aproximadamente a medio centímetro cuadrado de superficie.

Luego, esa imagen reducida es copiada varias veces sobre una lámina de película del mismo tamaño que la oblea.

Cada oblea limpia se recubre con una emulsión fotosensitiva para ser luego expuesta a la luz, mediante la primera máscara fotográfica. Esta permite que la luz llegue parcialmente a la oblea, endureciendo la emulsión de la zona iluminada. El dibujo de las áreas expuestas en la oblea suele ser tratado de diferentes formas. Generalmente, el plano que debe llevar al chip se va imprimiendo por capas.

Una vez finalizada la impresión de todas las capas, se sumerge la oblea en una solución que disuelve la zona de la emulsión que no se ha expuesto a la luz.
La capa superior consiste en hojas y tiras metálicas que son conectadas a las "patas" del chip. Estas "patas" comunicarán al poderoso componente de silicio con el mundo exterior.
Las reglas de esta comunicación se basan en "compuertas lógicas". Es decir que a cada entrada de dato específica, se responde sólo de una forma predefinida.
A las compuertas lógicas las forman las capas de caminos, las conexiones químicas y los puentes.
Las miles de compuertas lógicas instaladas dentro de un microprocesador, es decir, dentro del controlador central de una computadora y su chip de cálculo, permiten la llegada de datos en forma de impulsos electrónicos, además de alimentar al chip, procesar los datos o enviarlos a otros periféricos. En otras palabras, estas compuertas lógicas son las que permiten generar la mayoría de las operaciones que realizan las computadoras.
En el interior de los chips hay microscópicos componentes electrónicos como transistores, resistencias y diodos.

VERIFICACION DEL BUEN FUNCIONAMIENTO DEL CHIP

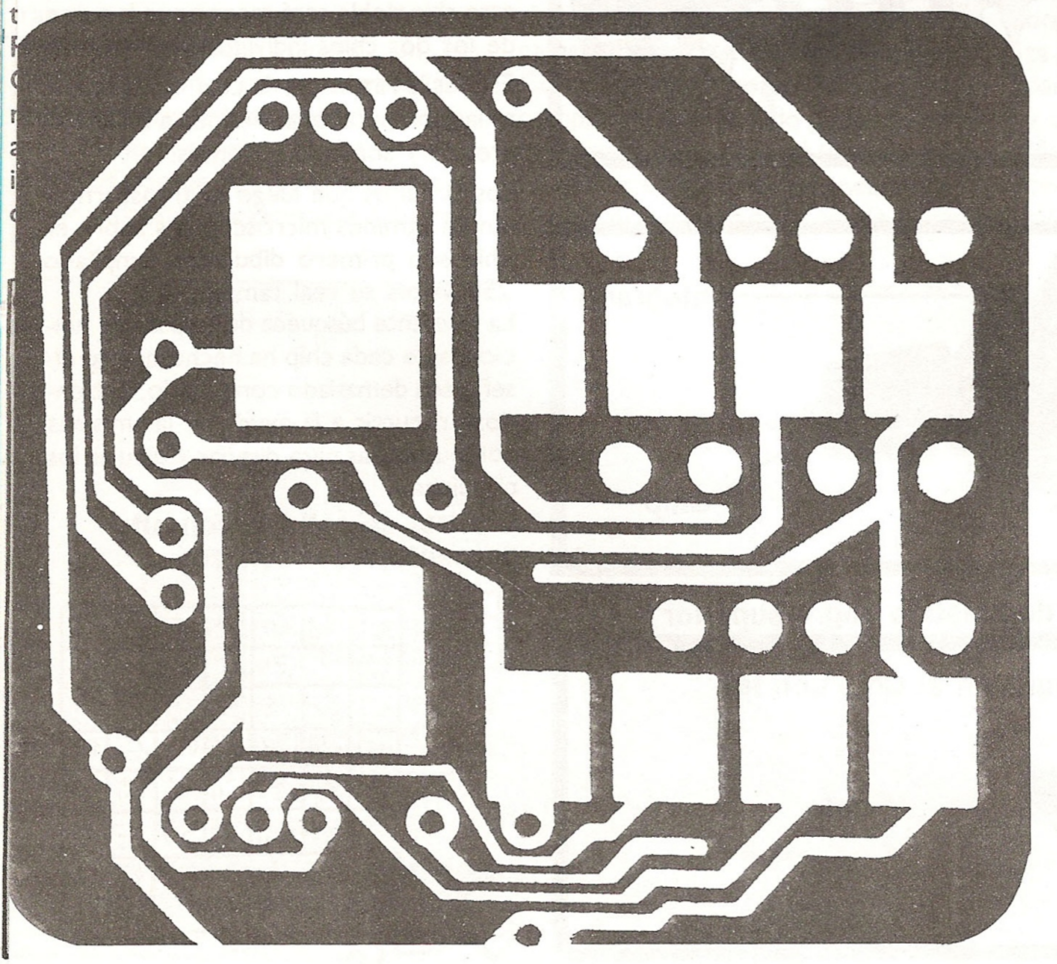
Cada una de las obleas contiene alrededor de cien futuros chips. Pero antes de ser empaquetados dentro de los integrados, cada chip es revisado para eliminar los defectuosos. Esta es una manera de de asegurar la fabricación de equipos sin fallas.

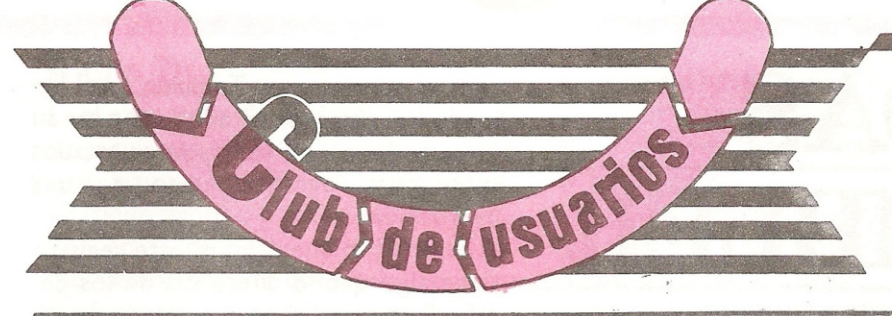
Para esto, la oblea se sumerge por una serie de sondas con agujas que se introducen en cada chip. Estos se conectan con un equipo encargado de testear el correcto funcionamiento del chip.

El chip que no funcione correctamente será marcado. Aquéllos que pasen la prueba de control, serán separados y empaquetados en cajas de plástico negras con conectores o "patas metálicas" conocidos con el nombre de "integrados".

CIRCUITOS IMPRESOS

Los chips que funcionan correctamente son conectados con otros componentes. El funcionamiento en conjunto de estos elementos electrónicos dentro de una computadora hace que el sistema funcione.
Para comunicar los chips y demás componentes, se fabrican las plaquetas impresas.
El mecanismo de la fabricación de estas plaquetas es similar al de la fabricación de los chips.
Se trata de comunicar a través de caminos de cobre, apoyados sobre una tabla rígida que está revestida en ambos lados por una delgada capa de cobre.
Se diseña el plano de cada lado de la tabla del circuito, considerando las conexiones que debe tener el chip. Luego, los planos son impresos sobre la plaqueta.
Se obtienen entonces los caminos comunicadores necesarios, sin que se crucen unos con otros. Por esto, el diseño de las plaquetas impresas es complicado y los fabricantes optaron por dejar esta tarea a las computadoras.
De esta forma, se determina la posición exacta de cada componente que posteriormente se apoyará sobre la plaqueta, como por ejemplo los chips.
A partir de este punto, el chip pasa a compartir entonces una plaqueta con diversos componentes como resistencias, capacitores, etcétera, que deberán funcionar en conjunto, por ejemplo, dentro de una computadora.
Lo curioso es que un chip diseñará otro que superará su potencia y a su vez éste último optimizará a otro chip, en una cadena infinita.





SETEO DE IMPRESORA ZENITH

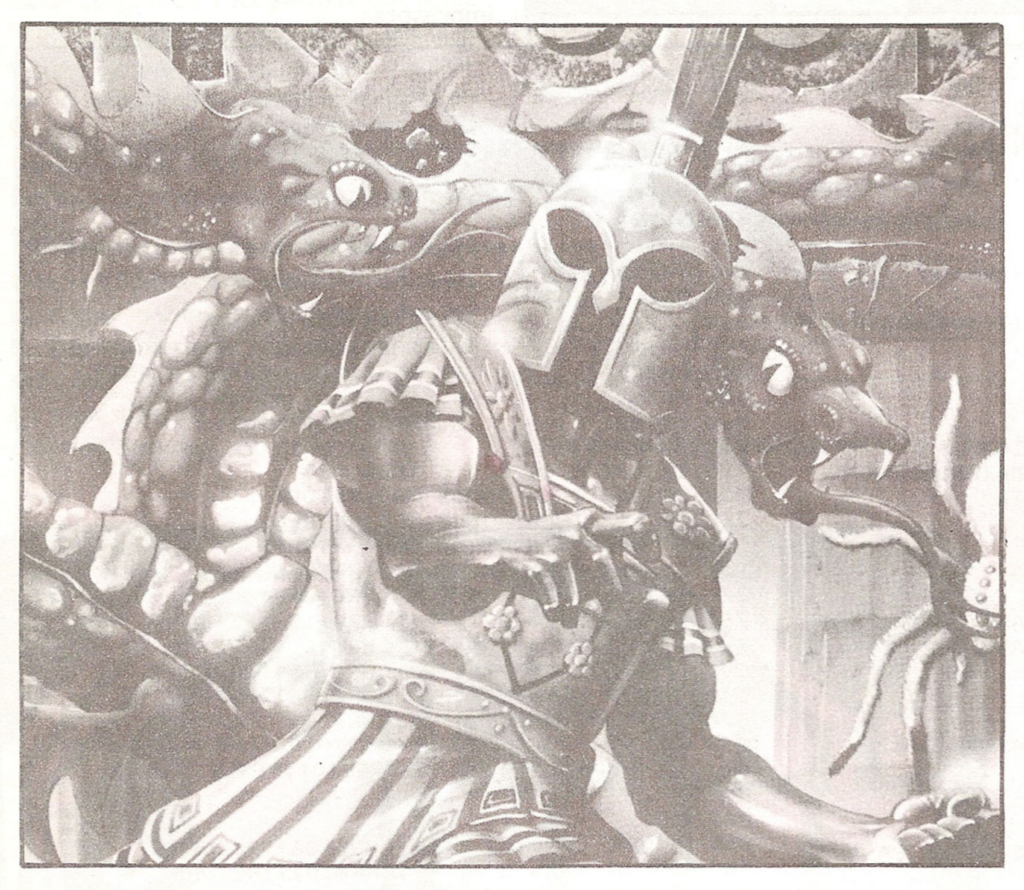


El programa que se edita a continuación (realizado en nuestro club), efectúa el seteo de una impresora Zenith, permite que el usuario pueda trabajar con diferentes tipos de letras (negritas, expandidas, comprimidas y otras).
Además este programa le otorga a la persona que lo utilice la posibilidad de elegir el espacio de líneas para una impresión, acorde con sus necesidades.
Para el mensaje de este programa no se requieren grandes conocimientos de computación, ya que sus pantallas son claras y sencillas de entender.
En otros números les entregaremos programas para seteo de otras impresoras.

```
10  KEYOFF
20  ON STOP GOSUB 1070:STOP ON
30 POKE &HFF89,&HDD:POKE &HFF8A,&HE1
40 KEY 10,"POKE &HFF8A,C9"+CHR$(13)
50 CLS
60 WIDTH 38
70 PRINT"***********************************"
80 PRINT"*                                 *"
90 PRINT"*         OPCIONES DE IMPRESION   *"
100 PRINT"*                                *"
110 PRINT"*********************************"
120 PRINT
130 PRINT
140 PRINT
150 PRINT"      1.LETRA NORMAL"
160 PRINT"      2.LETRA INTERNACIONAL"
170 PRINT"      3.LETRA CONDENSADA(REDUCIDA)"
180 PRINT"      4.LETRA EN NEGRITA"
190 PRINT"      5.LETRA 'ELITE'"
200 PRINT"      6.OTRAS OPCIONES
210 PRINT"      7.LETRA EXPANDIDA
220 PRINT"      8.PRUEBA
230 PRINT"      9.CONDENSADA EXPANDIDA NEGRITA"
240 PRINT"     10.CONDENSADA EXPANDIDA"
250 PRINT"     11.SALIR AL BASIC
260 LOCATE 0,20:INPUT A
270 IF A < 1 OR A >11 THEN 260
280 ON A GOSUB 290,300,310,410.430.540.820.
    830,860,850.840:GOTO 50
290 LPRINT CHR$(27)"@":RETURN
300 CLS:GOSUB 880:RETURN
310 CLS:PRINT"CONDENSADO COMUN O DOBLE PASADA"
320 PRINT"    (C/D)"
330 INPUT B$
340 IF B$ = "C" THEN GOSUB 370:RETURN
350 IF B$ = "D" THEN GOSUB 390:RETURN
360 GOTO 330
370 LPRINT CHR$(27);"!";CHR$(4)
380 RETURN
390 LPRINT CHR$(27);"!";CHR$(20)
400 RETURN
410 LPRINT CHR$(27);"!";CHR$(16)
420 RETURN
430 CLS
440 PRINT
450 PRINT
460 PRINT
470 PRINT"'ELITE' COMUN, DOBLE PASADA"
480 INPUT C$
490 IF C$ = "C" THEN GOSUB 520:RETURN
500 IF C$ = "D" THEN GOSUB 530:RETURN
510 GOTO 480
520 LPRINT CHR$(27);"!";CHR$(1):RETURN
530 LPRINT CHR$(27);"!";CHR$(17):RETURN
540 CLS
550 WIDTH 40
560 PRINT
570 PRINT
580 PRINT
590 PRINT"***************************************"
600 PRINT"*                                     *"
610 PRINT"*         TIPOS DE ESPACIADO          *"
620 PRINT"*                                     *"
630 PRINT"***************************************"
640 PRINT
650 PRINT
660 PRINT"          1.ESPACIADO NORMAL"
670 PRINT"          2.ESPACIADO INTERMEDIO(1 CM)
680 PRINT"          3.ESPACIADO GRANDE(2 CM)"
690 PRINT"          4.ESPACIADO SUPER(3 CM)"
700 PRINT"          5.ESPACIADO MINI"
710 INPUT B
720 IF B < 1 OR B > 5 THEN 710
730 IF B=1 THEN GOSUB 780:RETURN
740 IF B=2 THEN GOSUB 790:RETURN
750 IF B=3 THEN GOSUB 800:RETURN
760 IF B=4 THEN GOSUB 810:RETURN
770 IF B=5 THEN GOSUB 870:RETURN
780 LPRINT CHR$(27);"A";CHR$(15):RETURN
790 LPRINT CHR$(27);"A";CHR$(30):RETURN
800 LPRINT CHR$(27);"A";CHR$(45):RETURN
810 LPRINT CHR$(27);"A";CHR$(60):RETURN
820 LPRINT CHR$(27);"!";CHR$(32):RETURN
830 FOR Z=1 TO 10:FOR I=65 TO 90:LPRINTCHR$(I)
  ;:NEXTI:NEXTZ:LPRINTCHR$(10):RETUR
N
840 END
850 LPRINT CHR$(27);"!";CHR$(36):RETURN
860 LPRINT CHR$(27);"!";CHR$(52):RETURN
870 LPRINT CHR$(27);"A";CHR$(8.999999!):RETURN
880 CLS:WIDTH 40
890 PRINT"***************************************"
900 PRINT"*                                     *"
910 PRINT"*          1.LETRA INGLESA            *"
920 PRINT"*          2.LETRA CASTELLANA         *"
930 PRINT"*          3.LETRA ITALICA            *"
940 PRINT"*                                     *"
950 PRINT"***************************************"
960 LOCATE  0,13:INPUT SS
970 IF SS=1 THEN GOSUB 1010:RETURN
980 IF SS=2 THEN GOSUB 1030:RETURN
990 IF SS=3 THEN GOSUB 1050:RETURN
1000 GOTO 960
1010 LPRINT CHR$(27);"R";CHR$(10)
1020 RETURN
1030 WIDTH 38:CLS:PRINT "PARA ESE TIPO DE
   LETRA DEBERA TENER EL PROGRAMA CORRESP
ONDIENTE":FOR ZZ=1 TO 1000:NEXT ZZ
1040 RETURN
1050 LPRINT CHR$(27);"R";CHR$(19)
1060 RETURN
1070 RETURN
```

LAS RIENDAS DEL ASSEMBLER

Saber el significado de cada registro de este lenguaje es importante si no se quiere limitar la capacidad de trabajo. Por eso los explicamos uno por uno, y entre ellos el encargado de mostrarnos qué ocurre con nuestro conocido Z-80.



IX e IY:

Estos son los llamados registros índices y ya veremos porqué.
Su capacidad de almacenamiento es de 16 bits por cada uno y al igual que el PC o el SP, éstos tampoco son separables.
Tanto IX como IY cumplen con las características de los registros como el BC, o sea que se comportan como variables comunes. La diferencia entre éstos y el resto es la aplicación que se les da. En su momento veremos cuál es. Pero podemos decir que en Assembler, en contraposición con el Basic, sólo algunas variables pueden usarse con un determinado grupo de instrucciones.

Por ejemplo, las instrucciones de índole aritmética se realizan mayormente con el registro A (acumulador) como centro de programación.
Como vemos todos los registros son importantes en este lenguaje, dado que si evitamos el uso de alguno de ellos, nos estaremos autolimitando el grupo de instrucciones con las que podremos trabajar.
Por último veamos uno de los registros encargado de mostrar los que ocurre con el Z-80 paso a paso.

EL FLAG

Flag (del inglés bandera) es el registro F, y, como su nombre lo indica, es el encargado de la señalización de determinados efectos que ocurren al realizar ciertas operaciones. También es el llamado "registro de estado", por ser su tarea la de marcar los distintos estados o efectos logrados luego de una instrucción.
A pesar de lo que dijimos, este registro tiene algo que lo diferencia de los demás. Esto es así pues, a pesar de ser un registro de 8 bits como capacidad de almacenamiento, cada uno de sus bits tiene un significado diferente. Viendo el cuadro de la figura 1, estudiemos cada uno de los bits que nos han de interesar sin complicar demasiado aún las cosas:

C (bit 0):

Este al igual que cualquier otro bit de cualquier byte, como seguramente sabemos, puede tomar sólo dos valores: 0 ó 1.
Si este valor es uno el flag nos está indicando que de alguna forma se ha tratado de llenar el contenido de algún byte con un número superior a su capacidad de almacenamiento, o sea con un número superior a 255.
En cambio, si esto no ocurre, o lo que es lo mismo, el número en cuestión no rebasa la capacidad de un byte, el valor de este bit será 0.
Así, cuando este bit se encuentra en un uno lógico se dice que se produjo un acarreo en la operación o instrucción inmediatamente anterior.
Este es el origen de su nombre Carry (acarreo).

Z (bit 6):

Este es el llamado bit de Cero (Zero) que es el apropiado para avisarnos cuando el resultado de la operación o instrucción anterior fue cero.
Como todos los bits de Flag, su forma de indicar que ha ocurrido lo que él es capaz de indicar, es poniendo en 1 su contenido.
De esta forma, si el valor en algún momento es 1, nos estará indicando que el resultado de la operación anterior fue cero. Contrariamente, si el valor de este bit es cero, entonces el resultado de la operación anterior habrá sido distinto de cero. Como vemos, éste trabaja en forma contraria a lo que a primera impresión se podría suponer.
Recordemos que para un microprocesador el uno es lo que para nosotros la palabra "SI", y el cero la palabra "NO".
Retomando lo anterior podremos interpretar, que si Z es un uno, nos está queriendo indicar: "Sí, la instrucción anterior dio como resultado un cero".

S (bit 7):

Es el encargado de decirnos si el número resultante de la operación anterior tenía signo o no, es decir si era negativo (tiene signo) o positivo (no tiene signo), en "Complemento a Dos".

No desesperemos que todo tiene su explicación.

Un número como los que conocemos puede tener signo o no, pero en realidad esto es indiferente para el Z-80. Esto es así porque él no puede distinguir entre unos y otros, a los fines de realizar, por ejemplo, una suma algebraica. Además, tratemos de pensar cómo cuernos ponemos un "más +" o un "menos -" dentro de un byte. Aparentemente es imposible.

Hasta donde sabemos, los números que podemos poner dentro de un byte son siempre los mismos, de 0 a 255, y esto es así por lo que vimos antes de la capacidad que nos ofrecen esos 8 bits que el byte posee.

Y es cierto, los números serán siempre los mismos, lo que cambiaremos es la forma de interpretarlos.

Por ejemplo, ahora pensaremos que nuestro byte posee sólo 7 bits y no 8, que el bit sobrante es un regalo para poner allí su signo.

De esta forma, los bits 0 a 6 estarán como siempre destinados a almacenar un número, y el bit 7 u octavo (recordemos que éstos se numeran de 0 a 7), estará dedicado a almacenar el signo, ¿Cómo?. Si este bit es 1 (uno) nos estará indicando que el número siguiente tiene signo, en otras palabras, que es negativo. Lo contrario, por supuesto signficará que es positivo.

Pero como el astuto lector ya estará suponiendo, se ha limitado la capacidad de almacenamiento del byte, pudiendo escribir en él sólo números entre 128 y -127.

Volviendo al tema que comenzamos a tratar, el bit S del registro F nos indicará si el bit 7 del resultado de la operación anterior estaba en uno, poniéndose él también en el estado de éste último.

Es conveniente, como se habrán dado cuenta, tomarse esto con mucha, pero mucha calma, releyendo las veces que sea necesario cada párrafo para comprender poco a poco este alocado mundo.

Figura 1

B7	B6	B5	B4	B3	B2	B1	B0
S	Z	X	H	X	P/V	N	C

Figura 2

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
1	0	1	0	0	0	0	0

Figura 3

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	1	1	0	0	0	0	0

Figura 4

Número decimal	Representación en 8 bits							
	B7	B6	B5	B4	B3	B2	B1	B0
7	0	0	0	0	0	1	1	1
6	0	0	0	0	0	1	1	0
5	0	0	0	0	0	1	0	1
4	0	0	0	0	0	1	0	0
3	0	0	0	0	0	0	1	1
2	0	0	0	0	0	0	1	0
1	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0
−1	1	1	1	1	1	1	1	1
−2	1	1	1	1	1	1	1	0
−3	1	1	1	1	1	1	0	1
−4	1	1	1	1	1	1	0	0
−5	1	1	1	1	1	0	1	1
−6	1	1	1	1	1	0	1	0
−7	1	1	1	1	1	0	0	1
−8	1	1	1	1	1	0	0	0

$(123)_{10} = (01111011)_2$ su complemento es $(-123)_{10} \cong (10000101)$

ALGO MAS SOBRE LOS NUMEROS CON SIGNO...

Veamos la figura 2. Allí está representado un byte genérico, conteniendo un determinado número.

Si nos proponemos averiguar su significado decimal, veremos que éste es el 160 (podemos usar el comando ?&B10100000 del Basic MSX para hacerlo).

Pero si pensamos este número como uno capaz de tener signo, o sea en complemento a dos, vemos que su bit 7 está en uno. Por lo que podemos afirmar que es un número negativo. Lo que no sabemos es cuál es su valor exacto. Veamos cómo se piensa esto.

A todos los números positivos se les puede calcular su complemento.

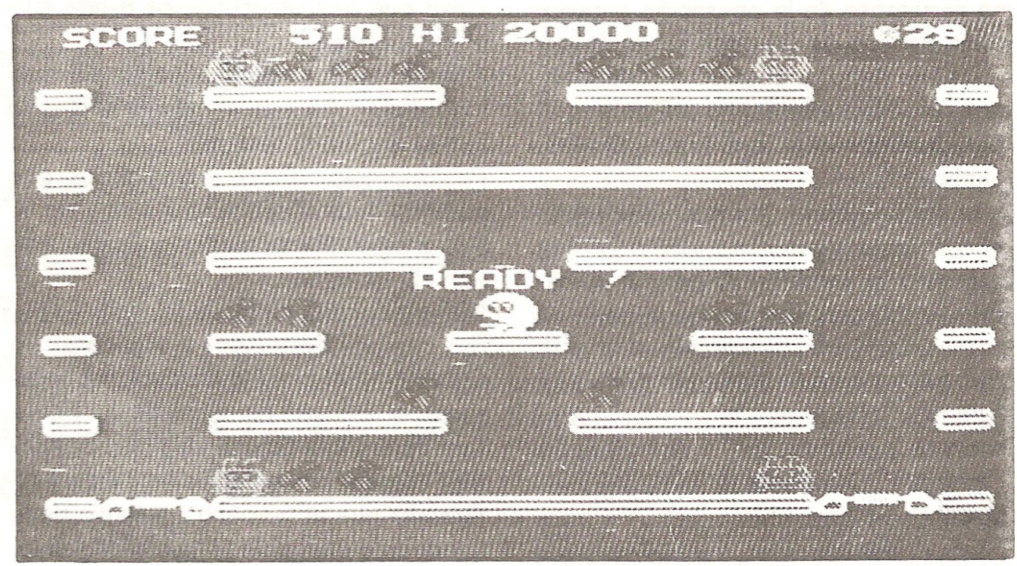
Por ejemplo, el complemento de 123 es —123. El tema es calcular este en binario, o mejor dicho, el problema se presenta al intentar representarlo en binario.

Para hacerlo veamos un método no-algebraico, que nos facilitará las cosas. Veamos nuevamente el byte de la figura 2. Para hallar su complemento, debemos leer el byte de derecha a izquierda, hasta encontrar el primer uno, salteándonos los ceros que estén antes de éste. Sin tocar a ese primer uno, cambiamos por su opuesto el valor de los siguientes bits, es decir, que donde haya uno 1 ponemos un cero y viceversa. Usando esto con el byte de la segunda figura, obtendremos el byte de la figura 3. En donde a simple vista, y pensándolo como un número en complemento, éste toma el valor 96, así que el número original era -96.

Repasemos atentamente este método, y veamos para corroborar esto la tabla de la figura 4.



FRUIT PANIC

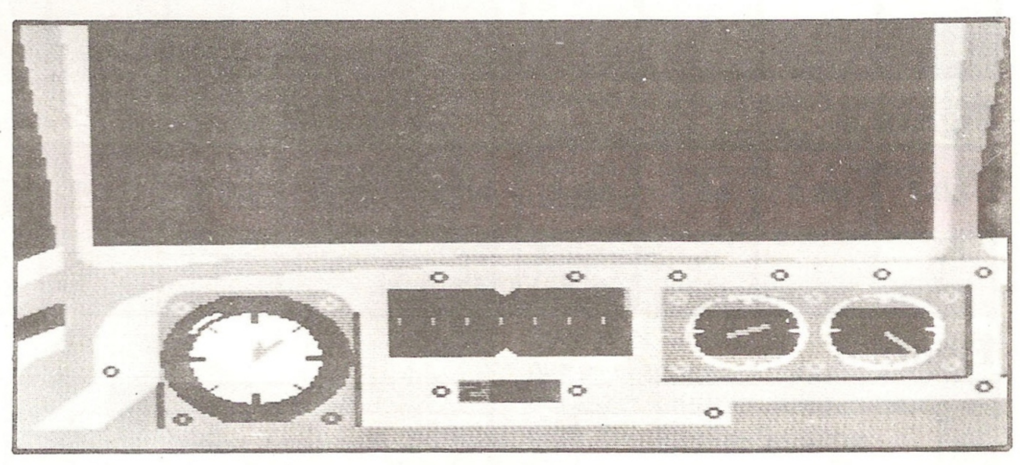


CREATIVIDDAD: 7
PRESENTACION: 7
ATRACCION: 9
GRAFICOS: 8
SONIDOS: 8
TIPO: ENTRETENIMIENTO
PRODUCE: MICROBYTE

Para los que ya han superado el legendario Pac Man, pero añoran ese gancho y furor que produjo, Fruit Panic será una buena opción.
Una suerte de gusanos o vaya uno a saber qué, ahora serán nuestros enemigos, y el laberinto se parece más a una estantería, en cuyos laterales inferiores, rebotaremos plácidamente para llegar a otro estante. Así comenzará la gran persecución contra este familiar cercano del Pacman, al que representaremos. Por supuesto, como es la costumbre de esta familia de redonditos, nuestra comida preferida será la plaga de frutas de los estantes.
También tendremos algún bocadillo que cada tanto podremos disfrutar. Y por último podremos tender nuestras redes para que,al pasar por ellas, nuestros enemigos queden atrapados.

THE DAM BUSTERS

CREATIVIDAD: 8
PRESENTACION: 8
ATRACCION: 10
GRAFICOS: 9
SONIDO: 8
TIPO: SIMULACION
PRODUCE: MICROBYTE



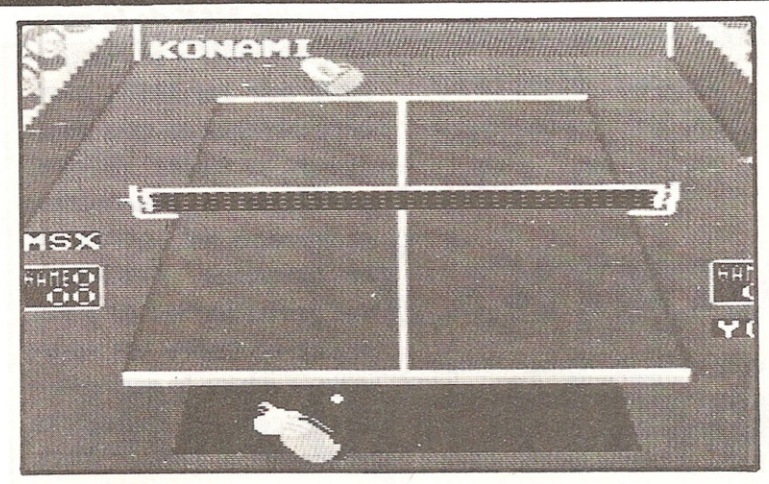
El sueño de pilotear un magnífico LANCASTER MK III —avión de combate— se hará realidad en este juego.
En efecto, todo —desde pilotear hasta ser el jefe de una escuadrilla de combate— podremos representar pasando por innumerables variantes y combinaciones.
Así, controlaremos la nave convertida en bombardero como los expertos ingenieros de vuelo, viendo con muy buenos gráficos todas y cada una de las pantallas en las distintas situaciones.
El juego viene acompañado por cuatro páginas oficio de explicación de las distintas variantes, y hasta con los gráficos esquematizados de las pantallas y estructura del avión.
Realmente podremos sentir casi lo mismo que cualquier piloto militar, en situaciones similares a las que se viven en la guerra.

PING-PONG

CREATIVIDAD: 10
PRESENTACION: 9
ATRACCION: 10
GRAFICOS: 10
SONIDO: 9
TIPO: ENTRETENIMIENTO
PRODUCE: MICROBYTE

Hasta ahora esta versión de Ping Pong de la firma Konami, que también está disponible para otras conocidas computadoras, es la mejor que hemos visto.

Sin lugar a dudas, a pesar de ser un juego tradicional, antiguo y común, su versión para computadoras es tan real y eficiente que no queda mucha diferencia entre la realidad y este desarrollo.
Es realmente excepcional. Por demás está resaltar la calidad gráfica, el realismo del sonido.
Es realmente un gran trabajo, y los responsables del mismo demuestran un gran poder de observación ante los hechos naturales de las actividades del hombre, a la vez que demuestran su capacidad para plasmarlo en la pantalla de una computadora.





GHOSTBUSTER

CREATIVIDAD: 8
PRESENTACION: 10
ATRACCION: 10
GRAFICOS: 9
SONIDO: 10
TIPO: ENTRETENIMIENTO
PRODUCE: MICROBYTE

Aquellos que vieron la película no se sentirán defraudados en absoluto con esta versión de los "Caza Fantasmas", y los que no tuvieron el placer de verla, realmente se asombrarán.

Un juego sofisticado y atrapante, donde nuestra misión será eliminar todos y cada uno de los temibles fantasmas.

Este nuevo desarrollo, seguramente nos atrapará de principio a fin.

Deberemos, como dijimos eliminar o, mejor dicho atrapar a la mayor cantidad de fantasmas posible, tratando de que llegue la menor cantidad de estos espectros al templo de Zull.

Pero volviendo al principio, nos deslumbraremos de movida, al ver el excelente dibujo de presentación acompañado de una música de igual o mejor calidad, la de la película.

Podremos además, si es nuestro gusto, hasta cantar al compás de ella pues en la presentación se incluye la letra de la misma en cada compás del tema.

Muchas son las opciones que nos presenta Ghostbusters, y todas igualmente atractivas.



Para comunicarse con nosotros deben escribirnos a "Load Revista para Usuarios de MSX", Paraná 720, 5to. Piso, (1017) Cap. Federal.

SALTO DE LINEA

Listé el programa "Diseñador de pantalla", pág. 24 del segundo número y no entra, ... una duda... la línea 1090 salta a 1110, ¿es error de imprenta o falta la línea 1100?

Jorge Horacio Sardin - San Miguel

LOAD MSX

Revisamos el listado y no encontramos ningún error. Y efectivamente, de la línea 1090 salta a la 1110. Te aconsejamos revisar tu listado y si aún no encuentras el error, envianos los mensajes de error que te impiden ejecutar el programa.

UN MOTOR REBELDE

A) En mi máquina no funciona la instrucción MOTOR. Al conectar el correspondiente cable al grabador se detiene y resulta imposible hacerlo funcionar, inclusive con MOTOR ON. Desearía me indiquen la posible causa del problema ya que he cambiado de grabador y todo sigue igual.
B) Hay programas para obtener un reloj por pantalla (la revista K64 publicó uno en su número 13, de abril). No obstante, no se logra el funcionamiento ininterrumpido del mismo ya que se detiene cuando se conecta un periférico (Drive-Cassette). ¿Existe algún programa sin este inconveniente?
C) Sería importante que se dieran las principales diferencias entre la C-128 y las MSX.
D) Los que vivimos en el interior, tenemos interés en adquirir los distintos programas del club de usuarios, pero no tenemos los precios como tampoco la forma de comprarlos.

Ricardo R. Bazzano - Azul

LOAD MSX

A) Desconocemos el motivo a simple vista, pero daremos unos consejos. Conviene conectar todos los cables y si el grabador no funciona, mantener la tecla de "PLAY" presionada en el mismo, y entrar en la computadora la sentencia "MOTOR" (sin ON u OFF) presionando RETURN posteriormente.
Normalmente luego de estas operaciones, se escucha como un chasquido en la computadora, pero no hay que alarmarse, esto es correcto.
Si aún no conseguimos solucionar el inconveniente, es probable que el cable del grabador no se encuentre en buenas condiciones. Tal vez no hace contacto correctamente. Pidamos algún cable (de algún amigo) de otra MSX.
Cada vez que entramos la sentencia MOTOR deberemos oir el chasquido en la computadora, si esto no sucede, sería conveniente hacer revisar la máquina por el servicio técnico.
B) La Unidad Central de Proceso sólo puede ejecutar una operación al mismo tiempo, como leer el teclado y actualizar la variable TIME.
Pero al leer o enviar datos a otros periféricos como impresoras, grabadores y drive, la CPU interrumpe las actividades mencionadas anteriormente. Es decir, no se lee el teclado, ni se actualizan las variables y menos aún, se continua ejecutando un programa. Todo se interrumpe para dar prioridad a la transferencia de datos.
La mayoría de los programas de ese estilo emplea la variable del sistema TIME. Como aclaramos antes, el valor almacenado en TIME no se actualiza cuando nuestro ordenador se comunica con algún periférico y entonces el reloj parece atrasar o detenerse.

C) Tendremos en cuenta tu propuesta de compararlas.
Para ambas máquinas se encuentra disponible el dBASE y planillas de cálculo para uso comercial.
No hay gran diferencia entre las ventajas de una y otra computadora en aplicaciones comerciales. Los creadores de soft se encargan de proveer los programas utilitarios similares para todas las marcas, por eso es difícil establecer si existe alguna con mejores aplicaciones que la otra.
D) Lo mejor será escribir al club de usuarios que nos interese.
En nuestra revista figura la dirección del club de MSX.

COMPATIBILIDAD MSX

Aprovechando para comentarles un problema que tengo. Poseo una máquina MSX de la marca CANNON y es compatible por ser MSX, pero el problema es que cualquier cassette (juego, utilitario o educativo) comprado no entran en la máquina. Problema del grabador no es pues probé en todos los volumenes e hice lo que ustedes dicen en la revista número 4 (muy útil). Además lo probé con otro grabador, fui a todos lados de computación sobre MSX y nadie me pudo arreglar eso.
P.D.: en las Talent entraron los programas. Otra duda, me parece que abajo de los cassettes dice "para Talent MSX" ¿No será eso?

Fernando M. Bidolegui - Capital Federal

LOAD MSX

Las MSX deben ser compatibles especialmente en cuanto a soft.
Pero un detalle que debemos tener presente es que los programas que se venden en los comercios están grabados para utilizarse sólo con grabadores "mono".
Por más que tu equipo de audio sea excelente, no te servirá para cargar este soft comprado, si es estéreo.
Podrías enviarnos un cassette grabado con algún programa de prueba, utilizando tu grabador y tu computadora. Nosotros probaremos si entra en otra MSX. Te agradeceríamos si nos envias, además, la marca y detalles de tu grabador.
Entendemos que las MSX CANNON son compatibles con Talent.
El hecho de que en los cassettes diga TALENT, no excluye que también sirvan para otras MSX, porque ésta es la regla principal que todos los fabricantes de ordenadores MSX deben respetar.